Exploiting Alignment in Multiparallel Corpora for Applications in Linguistics and Language Learning

Thesis
presented to the Faculty of Arts and Social Sciences
of the University of Zurich
for the degree of Doctor of Philosophy

by

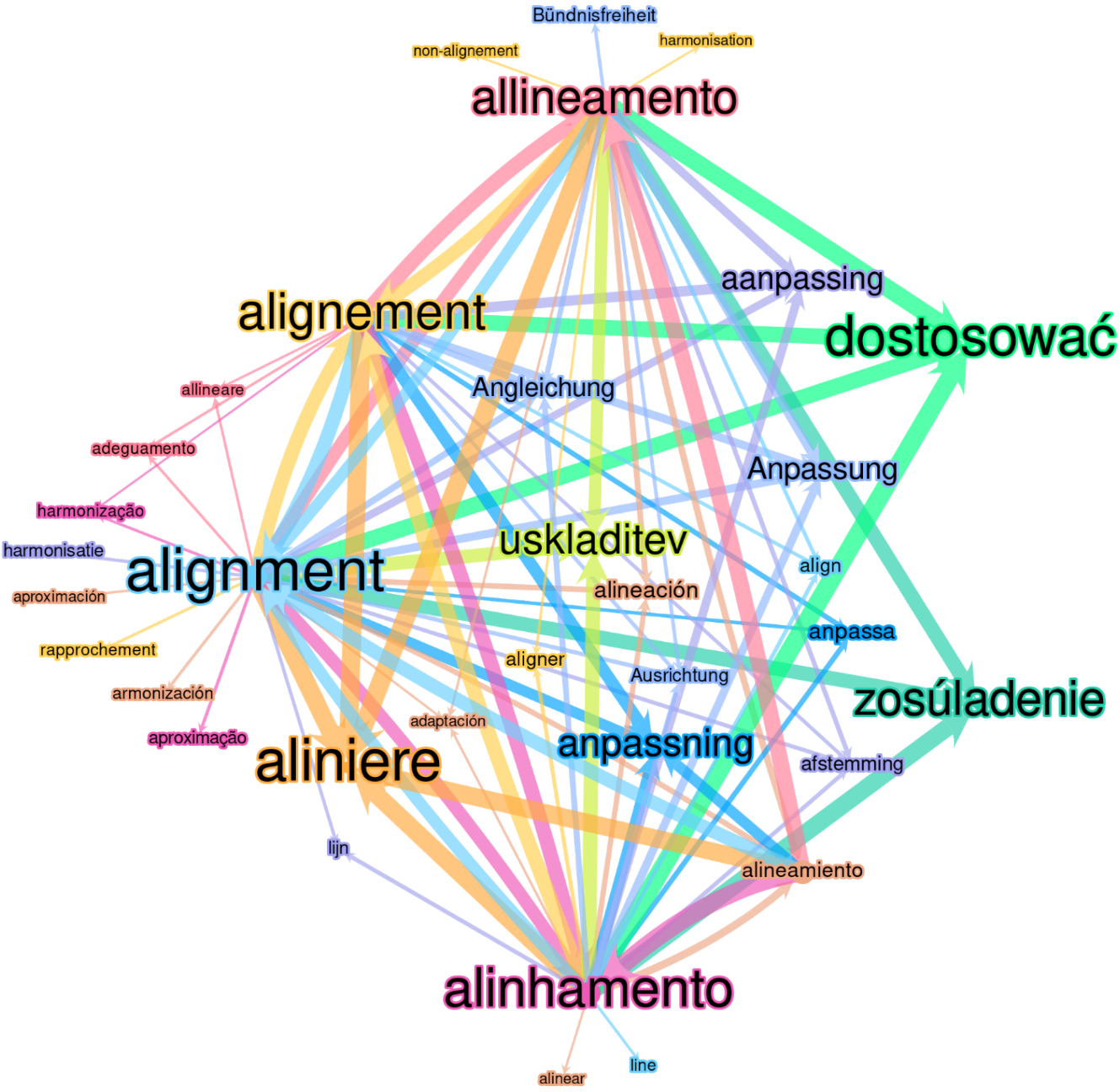
Johannes Graën

Accepted in the spring semester 2018
on the recommendation of the doctoral committee:

Prof. Dr. Martin Volk (main supervisor)
Prof. Dr. Marianne Hundt
Prof. Dr. Stefan Evert

Zurich, 2018

University of Zurich[UZH]


Bündnisfreiheit
non-alignement
harmonisation
allineamento
aanpassing
alignement
dostosować
allineare
Angleichung
adeguamento
Anpassung
harmonização
uskladitev
harmonisatie
alignment
align
alineación
aproximación
anpassa
rapprochement
aligner
Ausrichtung
zosúladenie
armonización
adaptación
aproximação
aliniere
anpassning
afstemming
lijn
alineamiento
alinhamento
line
alinear

Abstract

This thesis exploits the automatic identification of semantically corresponding units in parallel and multiparallel corpora, which is referred to as alignment. Multiparallel corpora are text collections of more than two languages that comprise reciprocal translations.

The contributions of this thesis are threefold:

- First, we prepare a large multiparallel corpus by adding several layers of annotation and alignment. Annotation is first performed on each language individually, while alignment is applied to two or more languages. For the latter case, we use the term multilingual alignment. We show that word alignment on parallel corpora can improve language-specific annotation by means of disambiguation.
- Our second contribution consists in the development and evaluation of prototypical algorithms for multilingual alignment on both sentence and word level. As languages vary considerably with regard to how content is realized in sentences and words, multilingual alignment needs to be represented by a hierarchical structure rather than by bidirectional links as prevailing representation of bilingual alignment.
- Based on our corpus, we thirdly show how word alignment in combination with different types of annotation can be employed to benefit linguists and language learners, among others. All tools developed in the context of this thesis, in particular the publicly available web applications, are driven by efficient database queries on a complex data structure.

Zusammenfassung

Gegenstand dieser Dissertation ist die Auswertung von semantischen Korrespondenzrelationen in parallelen und multiparallen Korpora, welche als Alignierungen bezeichnet werden. Multiparallele Korpora sind Textsammlungen wechselseitiger Übersetzungen zwischen mehr als zwei Sprachen.

Diese Arbeit umfasst drei Beiträge:

- Zum einen die Aufbereitung eines grossen multiparallelen Korpus durch Hinzufügen mehrerer Annotations- und Alignierungsebenen. Während jede Sprache zuerst separat annotiert wird, erstreckt sich die Alignierung über zwei oder mehr Sprachen. Letzteren Fall bezeichnen wir als multilinguale Alignierung. Wir zeigen, dass Wortalignierung in parallelen Korpora helfen kann, die sprachspezifischen Annotationen mittels Disambiguierung zu verbessern.

- Zum anderen die Entwicklung und Evaluierung prototypischer Algorithmen für multilinguale Alignierung sowohl auf Satz- als auch auf Wortebene. Aufgrund der starken Variation zwischen Sprachen bezüglich der Realisierung von Inhalt in Sätze und Wörter benötigt multilinguale Alignierung zur Darstellung der Korrespondenzen eine hierarchische Struktur, anstelle von bidirektionalen Verbindungen, wie sie bei bilingualer Alignierung üblich sind.

- Des weiteren zeigen wir, wie Wortalignierung in Verbindung mit verschiedenen Annotationsarten zum Nutzen u.a. von Linguisten und Sprachlernern eingesetzt werden kann. Allen Werkzeugen, die im Rahmen dieser Dissertation entwickelt wurden, insbesondere den öffentlich verfügbaren Webanwendungen, liegen effiziente Datenbankanfragen auf einer komplexen Datenstruktur zugrunde.

Acknowledgments

First of all, I wish to thank my supervisor Martin Volk who guided me through the initial troubles, gave me room to realize my own ideas and had the necessary confidence in me to finish this big project of mine. I am likewise indebted to my colleagues in the Sparcling project, Marianne Hundt, Simon Clematide and Elena Callegaro, and our student collaborators Dolores Batinic, Christof Bless and Mathias Müller.[1] Other, smaller contributions are indicated at the beginning of each chapter. I am thankful for Stefan Evert for accepting our invitation to become part of my doctoral committee.

During the last years, I had time to become acquainted with the other members of the Institute of Computational Linguistics. I really appreciate the supportive atmosphere that gave rise to interesting and fruitful discussions, often accompanied by chocolate and delicious pastry. I would like to give a special thanks to Noah Bubenhofer, Tilia Ellendorff, Anne Göhring, Samuel Läubli, Laura Mascarell, Jeannette Roth, Gerold Schneider and Don Tuggener. Aside from people in Zurich, I am particularly grateful for the Språkbanken group in Gothenburg for kindly receiving me in spring 2017.

The technical parts described in this thesis were quite demanding. My thanks therefore goes to our institute for making it possible to acquire new servers and letting me implement my concept for a new computer cluster that allows for distributed computing. This would not have been possible without the comprehensive support by the technicians of the Department of Informatics: Hanspeter Kunz, Beat Rageth and Enrico Solcà.

Los agradecimientos más importantes suelen venir últimos. Quisiera darles las gracias a mis amigos de Barcelona, particularmente a Alicia Burga, Graham Coleman, Gabriela Ferraro y Simon Mille, sin los cuales probablemente no hubiese empezado un doctorado. Les expreso mi gratitud a mi familia y a mi novia Mónica por el soporte incondicional durante todos estos años.

[1]The Sparcling project was kindly funded by the Swiss National Science Foundation under grant 105215_146781/1.

Contents

Chapter 1

Introduction

Collections of texts, known as text corpora, have been subject to linguists' interest for a long time. They served, for instance, lexicographers as a source for dictionary compilation or linguists and historians for the investigation of language change over time. **Parallel text corpora**, parallel corpora for short, sometimes also referred to as bitexts, are text collections in two or more languages where textual units, such as articles, sentences or words in one language correspond to textual units of the same kind in another language. If there is more than one other language, we will refer to these collections as **multiparallel corpora**.

The term parallel corpus covers merely translated material and not collections of texts that only connect to each other in terms of content. The latter ones are named **comparable corpora** since they describe the same topic in a comparable way without the necessity of texts being translations of each other. Wikipedia articles, as an example for comparable corpora, deal with the same topic in several languages and can either be translations from one or more existing articles, or be written independently of corresponding articles in other languages (for an overview see Plamada and Volk 2013).

The size of typical corpora impedes manual examination and, hence, calls for automatic processing. Natural language processing (NLP) deals with the automated treatment of natural language, predominantly in written form. NLP methods subdivide into **rule-based** and **statistical methods**. Approaches combining both paradigms are referred to as **hybrid methods**. Both types of methods are capable of processing large amounts of textual data in a tiny fraction of time of what a human would need to accomplish the same task. Automatic processing typically involves that some results are incorrect. While the main shortcoming of rule-based methods is coverage,[1] statistical methods bring about a task- and

[1] When dealing with natural language, there will typically be cases that the authors of the rules have not considered or that do not conform with the authors' intrinsic language model.

tool-specific **error rate**, that is, each partial result is expected to be incorrect with a known probability, but we do not know which parts are correct and which ones are not.

A principal motivation for developing new approaches is to achieve **lower error rates**. Some applications prefer to lower the error rate by excluding samples that are likely to include errors, other applications prefer large quantities of samples provided that the correct ones prevail. This trade-off between quality and quantity is known by the measures **precision** (how many of the results that we get are good?) and **recall** (how many of the good results do we actually find?) in binary classification tasks. A parametrization of the classifier that leads to an improvement of one measure usually implicates a decline of the other. The **F-Score** is a commonly used measure to account for both quality and quantity.

Applications that require a high precision and attach less importance to recall are typically concerned with **individual examples** and less so with statistics.[2] In **corpus linguistics**, in particular, these individual examples play a role when it comes to demonstrating the usage of particular word senses or expressions in context. Comprehensive dictionaries typically incorporate sample sentences for different word senses, which are oftentimes selected from corpora. A method for selecting those sentences, called **good dictionary examples** (GDEX) (Kilgarriff, Husák et al. 2008), ranks matching sentences according to features such as sentence lengths and rareness of the comprised words. Nonetheless, manual intervention is still needed to assess each dictionary candidate and sort out unsuitable ones.

Good sample sentences also play a role in **computer-assisted language learning** (CALL) applications. Some of them assist their users, which are language learners, by providing usage examples for a particular word or expression (just like dictionaries do) and can be used in automatically generated exercises. Although the completely automatic selection of sample sentences (see, for instance, Volodina et al. 2012; Pilán et al. 2016) always carries the risk of error (i.e., choosing a bad sentence, which confuses the learner instead of helping her), manual intervention is not feasible in such applications unless they rely on a list of precompiled exercises, which contradicts the principle of automating this task.

We address the inherent problem of errors in linguistic data that is processed with statistical methods by combining several layers of statistically generated data in parallel corpora, one of which typically is **word alignment** between the languages in question. Word alignment refers to the technique of automatically identifying corresponding words (i.e., the actual tokens) in corresponding sentences of different languages. Corresponding sentences, in turn, are automatically identified

[2] The contrary is the case for applications like machine translation, which learn generalized principles from large amounts of data. Errors, as long as they are not systematic, are simply smoothed out.

by means of **sentence alignment** techniques, for which the superordinate structures of sentences (paragraphs, articles, documents or, in general, texts) also need to be aligned. An erroneous alignment on any of these alignment levels necessarily implicates that subsequent alignments will also be wrong, that is, the **error propagates**.

While alignment is typically understood as identification and annotation of bilingual correspondences, we distinguish between bilingual and **multilingual alignment**, with the latter applying to cases where three or more languages are involved. As correspondence is a symmetric relation, multilingual alignments also require symmetry between all constituent elements. This symmetry is not guaranteed if we simply combine bilingual alignments of all language pairs.

The performance of alignment tools can be evaluated by means of comparison with a set of manually aligned data, a so-called **gold standard**. As for gold standards of any other kind of annotation, measures need to be taken to strive for consistency, which typically involves that the same annotation task is performed by multiple annotators and their results are compared. The **inter-annotator agreement** (for a comprehensive overview see Artstein 2017) indicates how well humans perform in a particular task.

1.1 The Sparcling Project

A large share of the corpus work in this thesis has been done as part of the Sparcling project.[3] The unabbreviated project name, "Large-scale Annotation and Alignment of Parallel Corpora for the Investigation of Linguistic Variation", does not only comprehend the two levels of corpus preparation that we are dealing with in this thesis, but it also pinpoints its objective: the investigation of linguistic variation.

One of the phenomena that was of particular interest to the linguists in our project is variable article use in English, that is, the usage or omission of articles. To investigate the factors that contribute to one or the other case, English is compared to several other languages. To render possible comparison between article use in English and another language, we need to know the correspondence between words and multiword expressions in both languages, which is approximated by means of automatically calculated word alignments. Additionally, syntactic information helps to identify relevant contexts.

The choice of languages with which to compare English is driven by typological differences. Article use is common in Germanic and Romance languages, while most Slavic and all Finnic languages do not possess articles. In the Sparcling

[3]The Swiss National Science Foundation supported our project under grant 105215_146781/1.

project, English was compared with German, French, Spanish, Italian, Polish and Finnish.[4] The Europarl corpus (Koehn 2005) was chosen as resource since it comprises, for the most part, transcripts of speeches delivered at the European Parliament's sittings in all these languages, and is thus arguably more suitable to model language use than, for instance, law or patent corpora.[5]

Several investigations on variable article use have been carried out with our corpora as basis. In (Callegaro et al. 2018), for instance, the corpus[6] is used to look up nominal phrases with two kinds of abbreviations: acronyms and initialisms. The meaning of acronyms in this study is restricted to abbreviating sequences of uppercase letters that are pronounceable; the unpronounceable variants are referred to as initialisms.[7] Corpus examples (approximately 300 examples per language) for a list of approximately 50 handpicked acronyms and initialisms were retrieved and manually analyzed. The main finding of this study is that articles are significantly more often used with (unpronounceable) initialisms than with (pronounceable) acronyms.

Other studies using extracts from our corpus are detailed in (Callegaro 2017). She also performed an analysis of how well the transcripts reflect what has actually been said at the sittings and finds that the transcripts are "considerably faithful to the speeches and can in turn be used for linguistic analysis."

1.2 Research Questions

This thesis deals with the preparation and exploitation of a large multiparallel corpus with annotations and alignments on different levels. It addresses the following questions:

1. What are the challenges in preparing a multiparallel corpus? Where does corpus preparation benefit from parallel data?

2. Which purposes can bilingual word alignment in multiparallel corpora be employed for?

3. How can we reliably determine multilingual alignments? Does concurrent alignment of more than two languages improve the quality of bilingual alignments?

[4] The project proposal envisaged Russian as representative of the Slavic language family, but it was later on replaced by Polish by reason of preference of the project participants.

[5] We give an overview of multilingual corpora in Chapter 2.

[6] This study uses version FEP6 of our corpus, earlier studies rely on FEP3. The properties of different corpus versions are explained in Chapter 3. The latest version of our corpus will be made available on the project website: http://pub.cl.uzh.ch/purl/sparcling_project

[7] In this thesis, we do not make this distinction and refer to both as acronyms.

A typical application for linguists of corpora is to query them. Querying means to retrieve matching examples for a given constellation (i.e., a combination of relations between tokens), which is described in terms of the underlying annotation layers. This aspect, though being an application on the final corpus, needs to be considered when building it. The first two points thus entail the question:

4. How can a large multiparallel corpus with several layers of annotations and alignments be queried efficiently?

1.3 Outline

In this chapter, we introduced the topics covered by this thesis and posed its principal questions. In the following chapters, we address them individually.

In **Chapter 2**, we deal with monolingual, parallel and multiparallel **corpora**. Existing corpora are described and classified in terms of token count and number of languages covered. We comment on the issues that we encountered in the Europarl corpus and considered detrimental to linguistic applications, and how we were able to resolve them for the most part in our corpus release.

Chapter 3 is concerned with corpus preparation and **annotation**. We detail the respective steps of tokenization, part-of-speech tagging and dependency parsing and point out the advantages that relational database management systems provide for corpus storage and querying.

Chapter 4 deals with existing approaches to **alignment** on the level of texts (or documents), sentences and words. We explain and evaluate our approaches to multilingual sentence and word alignment.

Chapter 5 describes four **applications** exploiting the rich structure of our corpora. We show how word alignment in connection with different annotation layers facilitates the creation of useful tools for corpus linguists and language learners.

In **Chapter 6**, we draw **conclusions** with regard to the initially raised questions and construe how **future work** can connect to the insights we gained.

Chapter 2

Parallel Text Corpora

The purpose of this chapter is to give the reader an overview about parallel and multiparallel corpora. Although there are numerous speech corpora available by now, this work only deals with **text corpora**. Every mention of the term 'corpus' has to be understood accordingly.

Before speaking about parallel corpora, we need to introduce **monolingual corpora** and their use cases first. In Section 2.1, we describe some applications that text collections have been used for in the past. In Section 2.2, we show use cases for **parallel corpora**, including bilingual subsets of multiparallel corpora, apart from training statistical machine translations systems, which still is the predominant beneficiary of parallel corpora. The last section, 2.3, is concerned with properties of **multiparallel corpora**. We describe the **Europarl corpus** and

Contributions

The cleaning and restructuring of the Europarl corpus (Section 2.3.1) would not have been possible without other people's contributions: Dolores Batinic performed the initial error analysis, Simon Clematide matched the respective speakers with the member list of the European Parliament and Mathias Müller wrote the XSLT rules for removing non-parallel data and adding additional speaker information for matching members.

All other tasks regarding the corpus preparation have been accomplished by the author. These tasks include the implementation of error correction rules (with manual examination of their respective effects on the corpus data), the alignment of speaker contributions in all available languages and error analysis on subsequent releases.

our **Corrected & Structured Europarl Corpus (CoStEP)** based thereupon, which, in turn, forms the basis of this work. Working with the Europarl data, we encountered several issues that we suspected to bring forth subsequent errors using standard natural language processing tools and thus pose problems for our envisaged applications.

Figure 2.1 depicts a number of **monolingual**, **parallel** and **multiparallel corpora** and their size with respect to the number of languages contained and the average size of tokens per language.[1] In some parallel corpora, the number of tokens is approximately evenly distributed over the comprised languages, in other cases their token number deviates considerably. The COPPA corpus (Junczys-Dowmunt et al. 2016), for instance, comprises approximately 160 million tokens in English and French but only 130 000 in Portuguese.

Östling (2015) presents a similar log–log representation of corpus sizes but includes the New Testament corpus (Cysouw and Wälchli 2007), which comprises 1142 translations in 1001 languages (Östling 2015). This discrepancy in numbers is due to several languages having more than one translation.[2]

The New Testament, though large in terms of languages, is small with regard to token size, which makes it inappropriate for particular applications that need reliable frequency estimations (e.g., machine translation (ibid.)). This is why we disregard all those numerous corpora that hold less than a million tokens per language, compilations of translated books, for instance.

Having been released half a century ago, the Brown Corpus (Kučera and Francis 1967) comprises American English texts with approximately one million token. The British National Corpus (BNC) (Leech 1992) with a hundred times more tokens and the Penn Treebank (Marcus et al. 1993) are notable English corpora from the '90s. 'Web as Corpus' (WaC) corpora (Baroni et al. 2009), obtained by **crawling web pages** with corresponding top-level domains (e.g., the top-level domain of Germany, 'de', for deWaC) were a new milestone in terms of sheer size. The similar-sized Annotated English Gigaword corpus (Napoles et al. 2012), which is based on the fifth version of the English Gigaword corpus (Parker et al. 2011), comes with several layers of annotation such as constituent parse trees and syntactic dependency relations. While manual work was included in the preparation of the Penn Treebank, the 1000 times bigger Annotated English Gigaword corpus relies solely on statistical annotation methods.

[1] For the overall number of tokens in these corpora, we rely on specifications given by the respective authors. If the number is given with the unit 'words', we assume them to be the same as tokens. For up-to-date numbers of more recent corpora, we consulted related web pages.

[2] The same is true for the OpenSubtitles corpus (Lison and Tiedemann 2016), which comprises two versions for Chinese (traditional and simplified) and Portuguese (the Portuguese and the Brazilian variant).

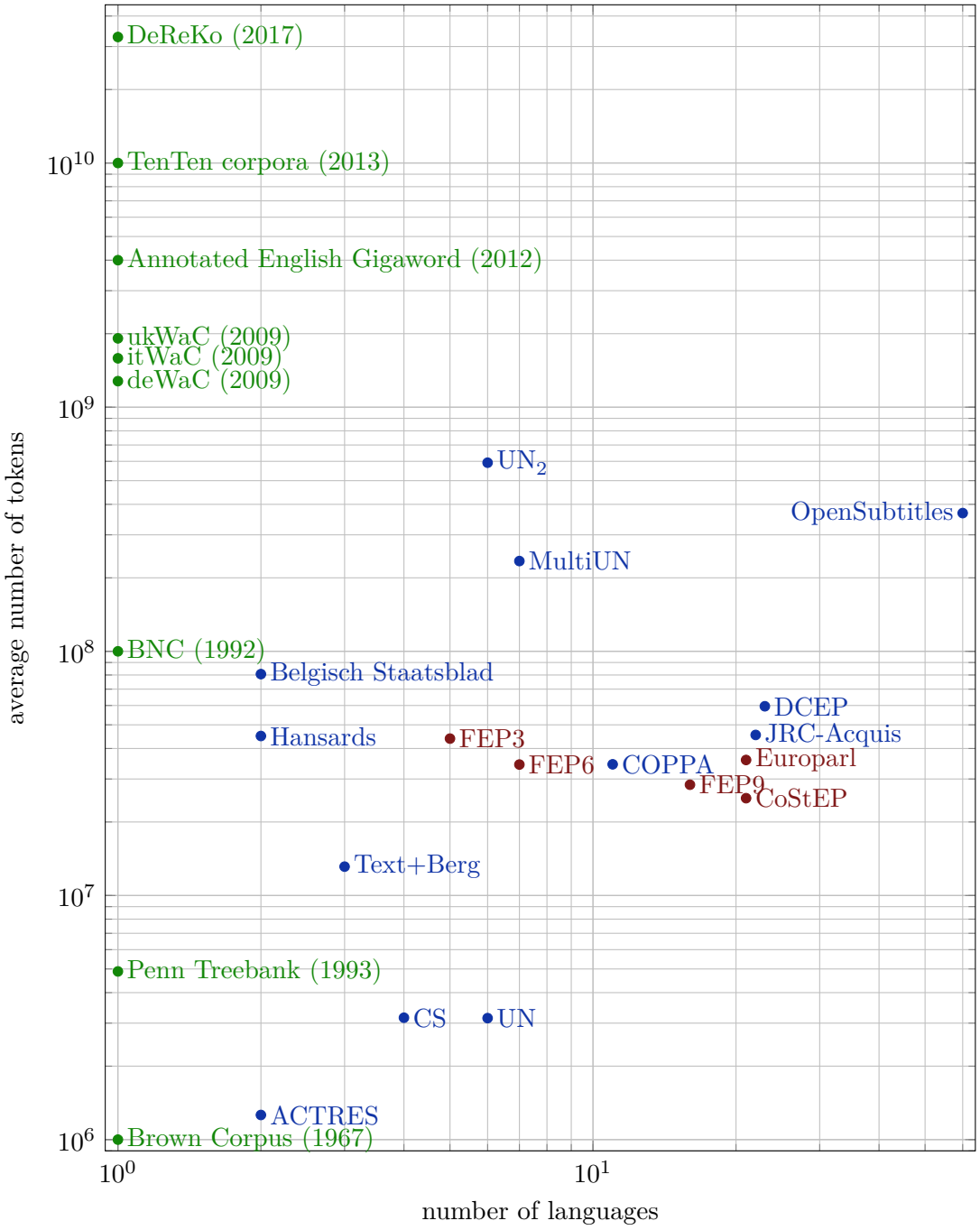


Figure 2.1 – Corpus size in terms of languages and average number of tokens. Monolingual corpora (green) have grown from one million (Brown Corpus, Kučera and Francis 1967) to more than 32 billion tokens (2017 version of DeReKo, Kupietz, Belica et al. 2010; Kupietz and Lüngen 2014). Parallel and multiparallel corpora are considerably smaller token-wise than recent monolingual corpora.

A collection of large corpora (Jakubíček et al. 2013), which holds more than 30 monolingual corpora obtained by crawling web pages, targets the size of 10^{10} tokens. That is why it has been named TenTen. The DeReKo corpus (Kupietz, Belica et al. 2010; Kupietz and Lüngen 2014) (the abbreviation translates to 'German reference corpus') aims at providing contemporary German texts from a wide range of text types. It currently comprises more than 30 billion tokens.

Kilgarriff (2001) proposes a strategy for objectively *comparing corpora* of the same language. Since web corpora comprise a huge number of different sources (i.e., the respective web pages crawled), we can imagine cases where an evaluation of a particular hypothesis confirmed by different web corpora actually relies on the same sources as they overlap.

A prototypical **parallel corpus** is the Canadian Hansards (described in Gale and Church 1991, 1993), which comprise the proceedings of the Canadian parliament in both English and French. Its token number approximates the number of tokens in Europarl (Koehn 2005) for languages that have been member of the European Union when the digital publication of parliamentary debates started in 1996. This is best shown in Figure 2.1 in comparison with the FEP3 corpus (see below), which only comprises those languages.

For comparison with other bilingual corpora, we added the Belgisch Staatsblad Corpus (Vanallemeersch 2010), a corpus comprising 10 years of Belgian governmental publications in Dutch and French, and the ACTRES corpus (Izquierdo et al. 2008), a compilation of English texts that have been translated to Spanish.

Large multiparallel corpora, that is corpora with more than two languages and more than one million tokens per language, predominantly originate from multinational organizations, which require all official documents to be translated into all their official languages. The United Nations (UN) have six official languages: Arabic, Chinese, English, French, Russian and Spanish. Both the UN corpus (Rafalovitch, Dale et al. 2009) and the MultiUN corpus (Eisele and Y. Chen 2010) are corpora obtained from texts published by the United Nations; the UN corpus from resolutions of the UN's general assembly and the MultiUN corpus from an archive of official documents. The latter corpus also contains a small part with translations into German, which increments the number of languages comprised by one. A more recent version of the UN corpus (UN_2) released by an official UN body (Ziemski et al. 2016) comprises a considerably large number of translations of any kind of UN documents in the official six languages.

From the European Union, which currently has 23 official working languages, several textual resources have been compiled to corpora. The arguably most prominent one, **Europarl** (Koehn 2005), comprises the debates of European Parliament over a period of 15 years. Our own corpus, **CoStEP**, provides a smaller portion of the texts contained in Europarl, but, in return, corrects several errors that we

identified in Europarl (see Section 2.3.1). We extracted several subsets of CoStEP for further processing the respective languages. The processing of these internal corpora (FEP3, FEP6 and FEP9) is detailed in Chapter 3.

Other corpora from the European Union include the JRC-Aquis (R. Steinberger, Pouliquen et al. 2006), a compilation of mostly legal texts, and DCEP (Digital Corpus of the European Parliament) (Hajlaoui et al. 2014), both compiled by the Joint Research Center (JRC) of the European Commission. These and other multiparallel corpora from the European Union are explained in (R. Steinberger, Ebrahim et al. 2014).

The by far largest parallel corpus with respect to both dimensions of size is OpenSubtitles (Lison and Tiedemann 2016), a corpus comprising user-translated movie subtitles from OpenSubtitles.org.[3] Due to the nature of movie subtitles (e.g., duration of visibility and space constraints), this text type arguably differs from all other multilingual corpora introduced above. Movie subtitles transcribe the speech in movie scenes, typically performing translation at the same time. The resulting sentences spread over several subtitle blocks that are shown sequentially to the viewer with a time distance of only seconds (see ibid., Section 3). This restriction presumably enforces the subtitle creators to avoid overly complicated sentences. The debates of the European Parliament in Europarl have also been transcribed and translated to other languages. In contrast to how we expect sentences in the subtitle corpus to be, the average sentence in our excerpt from Europarl consists of 24 tokens.

2.1 Monolingual Corpora

Linguists and scholars of other disciplines have employed collections of texts to gain insights into various aspects of (written) language. In human history, cryptographers made use of those text collections to derive – over time – increasingly complex statistics that helped them decipher encrypted messages (S. Singh 1999). The principle behind these efforts remained the same: If we know the language that is behind an encrypted message, we try to identify the same properties in its nonsensical textual representation and use them to align both 'languages'. Assumed that the language of the unencrypted message is English, we can start by comparing single letter frequencies. If the message is long enough and encryption has been done by exchanging letters,[4] we expect the most frequent encrypted letter to correspond to the letter 'e', the most frequent one in English, in the original

[3] http://www.opensubtitles.org/

[4] This is arguably the most insecure encryption method, though it was used successfully in ancient times. Gnxr, sbe vafgnapr, gur EBG13 fhofgvghgvba.

text. This method extends to the distributional comparison of letter n-grams, word forms and word n-grams, all of which can be obtained from a text collection, a text corpus.

While early cryptographers were mainly interested in statistical properties of languages with the objective of deciphering encrypted messages, linguists in the pre-computer era performed manual analyses on small text collections to find out how language works. Firth (1957a) uses the limericks of Edward Lear as corpus to illustrate his notion of collocation. The word 'man' in these Limericks, for example, shows a tendency to be preceded by 'old' and the sequence "old man" frequently appears in the phrase "There was an Old Man of" followed by the name of a place.[5] More recent works on collocations (e.g., Church and Hanks 1990; Michelbacher et al. 2007; Evert 2004, 2008; Gries 2013; Bartsch and Evert 2014) continue with this kind of analysis by applying statistical measures to substantially larger corpora.

Technical advancements in the second half of the last century rendered possible the compilation of large electronic corpora. Church and Hanks (1990) report that the development of "facilities for the computational storage and analysis of large bodies of natural language" allowed them to perform concordance analysis on various (English) corpora. The size of textual material contained by a corpus matters in terms of its representativeness. Other parameters such as text quality, coverage of domains and genres, and consistency of annotation, however, may be more important for particular use cases.

With the creation of the British National Corpus (BNC) (Leech 1992) in the early '90s, researchers intended to compile a resource representing British English (at that time) from a variety of perspectives (also including speech). Although targeting a broad number of applications and fields of research, Leech has the impression "that the uses to which a corpus can be put are far more numerous and varied than the corpus compilers could have imagined."

With the rise of the Internet, it seemed obvious to use the manifold texts on web pages around the world, the vast majority of them being in English, for compiling corpora that reflect a language in its breadth, which was actually the point of the BNC's 'corpus design' (ibid., Section 4.1). These web corpora, however, do not allow for controlling the exact amount of text for each category. They may also contain spelling errors or texts that only superficially look like the language they are supposed to comprise, but are outright wrong, in fact. Furthermore, automatic translations of texts, in particular concerning English as frequent target language, can bias our impression of a language when using web corpora as reference.[6]

[5] Which becomes apparent when looking only at the first lines of Edward Lear's limericks: http://www.bencourtney.com/ebooks/lear/#indexfirstlines

[6] This issue has been addressed by, for example, (Antonova and Misyurev 2011).

The 'web as corpus' (WaC) initiative (Kilgarriff and Grefenstette 2003; Baroni et al. 2009) concerns itself with theoretical and practical questions regarding the use of the Internet for corpus compilation. Three of the early corpus releases, deWaC, itWaC and ukWac, are shown in Figure 2.1. Successors such as the English Gigaword corpus (Napoles et al. 2012) or the TenTen corpus collection (Jakubíček et al. 2013), a series of web corpora compiled for many languages with a target size of 10^{10} tokens, have pushed the limit to new dimensions.

While those are corpora compiled by crawling the web, the German DeReKo corpus (Kupietz, Belica et al. 2010; Kupietz and Lüngen 2014) is aimed at the "documentation of the German language in its current use". Unlike the aforementioned web corpora, it consists of licensed material, that is, copyrighted material that has been added to the corpus with permissions of the respective rights-holder. The downside of that is that the corpus can only be made available to a dedicated interface that does not allow for reconstruction of the original texts.

Other corpora not detailed here include learner corpora, that is, collections of texts written by language learners including, on purpose, their mistakes, historical corpora, for instance, for investigating language change (Curzan 2009), and dialect corpora, which often comprise transcriptions of speech recordings (Anderwald and Szmrecsanyi 2009). Learner corpora are frequently annotated with the (or a) correct language use so that we can learn about the learners. In Section 5.4, we use learner corpora to verify if we can predict learner errors on the basis of our annotated and aligned corpus.

Corpus Query Languages

The number of corpus query languages for monolingual corpora is numerous. Clematide (2015) classifies a list of major query languages into a scheme derived from their most distinctive properties. There is no query language that outperforms others. The choice of query language depends foremost on how the corpus has been processed and which annotation layers are available; and these questions potentially depend on the query language that one plans to use.

Since we are mostly concerned with querying parallel corpora, we refer the interested reader to (S. Brants et al. 2002) for the description of the Tiger Treebank and its query language, (Meurers and Müller 2009), who show how to effectively employ this query language for linguistically motivated corpus searches and (Bański et al. 2016) for a corpus query standardization approach with the name 'corpus query lingua franca' (CQLF).

Efforts have been made to allow the access of corpus query systems (i.e., the applications that interpret the actual query and perform corpus searches) via web applications, either for a restricted or unrestricted number of users (depending on

copyright and for performance reasons). The flexible architecture of web applications facilitates a user interface design that supports non-expert users in performing corpus queries. An example of such a system is shown in Figure 5.1.

CQPweb (Hardie 2012) and ANNIS (Krause and Zeldes 2014) are other widely-used web applications for corpus search that make use of corpus query languages, but at the same time provide support for inexperienced users. The SketchEngine (Kilgarriff, Rychlỳ et al. 2004; Kilgarriff, Baisa et al. 2014), modestly described as "the ultimate tool to explore how language works", targets both groups: expert users (e.g., linguists or lexicographers) and non-expert users (e.g., teachers, students or historians).

The query syntax for attributes in CQP and ANNIS is very similar. For instance, the CQP query ⟨`[pos="DET"] [pos="ADJ"] [pos="NOUN"]`⟩ is used to retrieve sentences where a determiner (i.e., article) is followed by an adjective, which, in turn, is followed by a noun. The same query is expressed in ANNIS syntax by means of the precedence operator "." (⟨`pos="DET" . pos="ADJA" . pos="NOUN"`⟩), however, ANNIS is capable of querying arbitrary relations between tokens, which has been envisaged for CQP (Evert and Hardie 2015), but still lacks implementation.

2.2 Parallel Corpora

The origin of linguistic research on parallel texts dates back to archaeological excavations comprising bilingual or trilingual inscriptions like the Rosetta Stone, which is a stele with approximately the same text engraved in three different ancient languages.[7] The stone was discovered at a time when all three languages (Ancient Greek and two versions of the Egyptian language using different scripts, namely Demotic and hieroglyphic Egyptian) had been extinct for more than a millennium, but by means of bequeathed knowledge about Ancient Greek, scholars managed to gain comprehension about the other two as well.

More recently, international collaborations, whether cultural, political or economical, have generated an extensive amount of parallel resources, from bilingual ones up to several dozens of languages. These resources include legal texts, transcribed speeches, instruction manuals, package inserts for drugs and translated books, such as the Bible, textbooks or science fiction.

Many of those resources have already been exploited by linguistic applications, such as word sense disambiguation (Kazakov and Shahid 2013), learning syntactic translation rules between languages (Lavie et al. 2008), computational lexicography (Tiedemann 2003b) or machine translation (ibid.).

[7] This is actually an example for a multiparallel text.

The Canadian Hansards, the proceedings of the Canadian parliament, are made available in English and French. From the mentions in (Gale and Church 1991, 1993), we know that they had a sample of 90 million words in both languages, English and French, together. This number resembles the size of monolingual corpora from that time. Another, more recent and marginally bigger parallel corpus from a governmental body is the Belgian Staatsblad corpus (Vanallemeersch 2010), which comprises governmental publications in Dutch and French. In theory, we should be able to compile parallel corpora from official translations of any officially bilingual country's publications (for a list of other, little-known parallel corpora see Xiao 2008; Aijmer 2008, Chapter 6).

Corpus Query Languages for Parallel Corpora

In the previous section, we have touched on the topic of corpus query languages for annotated monolingual corpora. What all those query languages have in common is that they are represented as text with a formal grammar motivated by linguistic attributes and structures (Clematide 2015), with the consequence that they can only be used by expert users. Those expert users need to know the characteristics of each annotation layer (e.g., the tagset used for part-of-speech tagging) to be able to formulate a query.

In contrast, parallel concordancing systems facilitate the access to corpus material to non-expert users (Volk, Graën and Callegaro 2014). Their users do not need to learn a complex query language, but can perform corpus searches by simply typing in sequences of words. This comes at the price of expressivity. It is typically not possible to further restrict searches and filter out unwanted cases. When we are, for instance, interested in translations of the expression 'to level something at somebody' and we search for 'level at' or 'levels at', the system will also find cases where 'level' or 'levels' is a noun.[8]

The Stockholm TreeAligner (Volk, Lundborg et al. 2007; Lundborg et al. 2007) is a tool to visualize parallel treebanks and to allow its user to add or modify alignments between constituents of two parallel syntax trees. Furthermore, it includes a module for performing corpus searches with partial search queries in TIGER-Search syntax (König and Lezius 2000) and a further restriction on the alignment of the independently matched constituents.

[8]One of the systems that we examine in (Volk, Graën and Callegaro 2014), Linguee, finds "noise levels at night", "utilisation levels at waste incineration plants", "regional and local levels at the same time", "can reach levels at which", etc., but only two cases with 'level' as a verb: "despite all of the criticism leveled at Credit Suisse regarding the promotion of women" and "the criticism which the applicant levels at the findings relating to the need to keep basic geometric forms available".

Table 2.1 – Sample parallel query in TreeAligner syntax from (Lundborg et al. 2007). The query searches for aligned noun phrases (NP) in parallel treebanks with the additional restrictions that the noun phrase in the first treebank dominates an adjective phrase (AP), that the noun phrase in the second treebank dominates a prepositional phrase (PP), and that these two dominated phrases are aligned.

Part	Query
First treebank	`#np1:[cat="NP"] > #ap:[cat="AP"]`
Second treebank	`#np2:[cat="NP"] > #pp:[cat="PP"]`
Alignment	`#np1 * #np2 & #ap * #pp`

A sample query for the TreeAligner is given in Table 2.1. Since its treebanks use a phrase structure syntax, the standard relational query operators besides precedence (in addition to attributive restrictions on the structural elements) are the direct (“>”) and indirect dominance (“>*”). The same operators are used by ANNIS for queries on phrase structure syntax. ANNIS also allows for dependency syntax queries. The query ⟨`pos=/V.FIN/ ->dep[func="obja"] pos=/N.*/`⟩, for instance, searches for finite verbs (auxiliary, modal or main verbs) that have a noun (proper or common noun) as direct object.

2.3 Multiparallel Corpora

Most applications on parallel corpora work on one pair of languages at a time: a source and a target language. Some applications, however, employ a third language, called pivot or bridge language, to support operation between source and target language. This technique is called triangulation (see for instance Borin 2000a; Cohn and Lapata 2007; Bouma et al. 2008; Y. Chen et al. 2008). Triangulation thus requires a parallel corpus of at least three languages. In Section 5.1, we show how massive triangulation, that is, the simultaneous triangulation over all available pivot languages, compensates for potentially erroneous or missing word alignments in one of the third languages. We use a similar approach in Section 3.2.1 to disambiguate ambiguous lemmas.

In the majority of use cases for multiparallel corpora found in the literature, bilingual data is extracted and processed, potentially for many language pairs, but predominantly independent of other languages. This applies, in particular, to the field of machine translation. The OPUS (open parallel corpus) collection[9]

[9] http://opus.nlpl.eu/

(Tiedemann 2009, 2012) comprises a long list of freely available parallel corpora. In addition to the respective corpus as a whole, the collection also offers sentence-aligned subsets for language pairs.

Our institute has edited and published two multilingual corpora, which are based on written texts, in contrast to OpenSubtitles and Europarl (for the most part), are not formally restricted, unlike the JRC-Acquis, and deal with a wide variety of topics. The Text+Berg corpus (Göhring and Volk 2011) comprises the yearbooks of the Swiss Alpine Club for more than 150 years. These yearbooks are collections of articles written by the Alpine Club's member and deal with topics such as "climatology, geology, fauna, flora, society, culture, tourism, leisure and sports" (ibid.). Most of the articles are available in French and German, more recent yearbooks also cover Italian. The second corpus only spans 120 years of time. The Credit Suisse (CS) Bulletin Corpus (Volk, Amrhein et al. 2016) comprises the Credit Suisse banking magazine in English, French, German and Italian, but issues older than 1970 are only available in French and German. Although issued by a Swiss bank, a wide range of non-money-related topics is covered in the magazine.

2.3.1 Our CoStEP Corpus

At the time the Europarl corpus (Koehn 2005) was released, few multilingual resources were available. The author starts with saying that "progress in natural language research is driven by the availability of data", which we only complement by the technical advancement.[10] It is thus not surprising that the Europarl corpus became a reference for training and testing statistical machine translation systems (see, for instance, DeNero, Gillick et al. 2006; Søgaard and Kuhn 2009; Crego et al. 2010).

Besides machine translation, Koehn also mentions "word sense disambiguation, anaphora resolution, information extraction" as use cases. Other than that, Europarl has been used, for instance, for grammar projection (Bouma et al. 2008) unsupervised part-of-speech tagging based on projection between languages (Das and Petrov 2011) or "learning multilingual semantic representations" (Hermann and Blunsom 2014).

When we worked with the Europarl data, which is raw text, split into speaker contributions (turns) by meta information, we discovered several recurring errors that had a negative effect on natural language processing tools applied to it. Though not falling into the category of web corpora, the Europarl data (i.e., the minutes of the European Parliament's plenary debates) has been scraped from the

[10] Deep learning techniques (e.g., Collobert and Weston 2008) would certainly not have performed well on an "IBM Model 3090 mainframe computer with access to 16 megabytes of virtual memory" (Gale and Church 1993).

European Parliament's website. We assume that the minutes published on that website have been manually edited and that this editing accounts for some errors found.

The most frequent error we encountered in approximately every other turn is that either parts of the actual text have been categorized and marked as meta information, or meta information has not been recognized as such and forms part of the running text. An example for the former case is "`<SPEAKER ID="115" LANGUAGE="" NAME="" AFFILIATION="The Minutes of the previous sitting were approved.)"/>`", and for the latter "Miller (PSE). (EN) Herr Präsident, ich […]" or "(RO) Бих искала да поздравя г-н Stolojan за […]".

Other errors include a partially performed tokenization on all scraped texts, which works well for English, but is problematic for other languages. In total, we identified 11 types of errors and classified them in several dimensions (Graën, Batinic et al. 2014). On the one hand, we estimated the error frequency. For some errors, we simply count their frequency; for other errors, we can estimate it by extrapolation.[11] In addition to frequency, we also assess whether the error originates in the published web pages or is due to text processing by Koehn, and judge the impact of each error type to further processing pipelines.

Despite Europarl's application to various linguistic tasks, we believe that its use for corpus linguistic investigations is limited since the numerous errors we identified will have an impact on the quality of natural language processing tools applied to it. As errors tend to accumulate, a single error in the corpus input can lead to numerous errors in an application at the end of a processing pipeline.

The partial tokenization described above, for instance, confuses a standard part-of-speech tagger with included tokenization as preprocessing step, as we show in (ibid.). In case we decide to skip the tokenization step, we will, however, miss all the cases where tokenization has not been performed yet. A greater risk with regard to the utilization of parallel data from Europarl is the suggested alignment of texts, which is frequently misleading (see below). If we were to take all texts that appear at the same position in Europarl's parallel documents as translations of each other, our sentence and word alignment algorithms (Chapter 4) would learn correspondences from unrelated (i.e., non-parallel) texts and thus deteriorate the alignment models as a whole.

Our goal, to annotate and align the Europarl corpus with the objective of linguistic research, required us to correct the identified errors first and assure that

[11] A special error type is the omission of foreign words in other languages. Those words are formatted as italic text with HTML mark-up, but have not been marked in older versions. This is presumably the reason why they this case was not paid attention to and foreign words are thus removed from the text like any other mark-up.

supposedly parallel texts actually are corresponding translations. This led to the *Corrected & Structured Europarl Corpus (CoStEP)*.[12]

Knowing the different types of error, their frequency and impact, we aim at correcting them. To do so, we use regular expressions to identify errors, inspect the identified cases (manually or by aggregation) and modify the texts accordingly if we do not encounter false positives. We do this for all corrigible errors that we found.[13] We cannot, however, correct errors in cases where information has been lost (e.g., reinsert the omitted words).

Besides the cleaning part, we also align the respective speaker turns in all languages. In 58 % of the documents, which typically comprise a single day of plenary debates, the structure given by the meta information is the same in all languages and the alignment task is thus trivial. We are required to fuzzy match speaker names and other meta attributes in the remaining cases to extract corresponding turns. Alongside the consistent turn structure, we also distinguish between actual speaker contributions and comments in the minutes and mark every text parts as being either speech or a minutes' comment. Within the respective turns, we mark quotations since quotation marks have been inconsistently used (e.g., a double comma instead of the lower opening quotation mark in German). This allows a corpus user to replace the quotation mark-up with language-specific typographic or typewriter quotation marks, depending on what further text processing tools are able to handle.

A more recent addition, which is not explained in (ibid.), is the matching of speakers to a list of European Parliament's members. That way, we are able to add more reliable information to the metadata of each turn. Instead of just a name given in the original data, which either holds the respective speaker's full name or last name only (both frequently with different spellings or spelling errors), we add forename and surname from a reliable source. Additionally, we add the country and political group a member is representing from that list.

[12] The corpus is available at `http://pub.cl.uzh.ch/purl/costep`.

[13] This includes undoing the partial tokenization that has been applied to the texts.

Chapter 3

Corpus Annotation Methods

Our project research questions (see Section 1.1) demand a large **multiparallel corpus** with several layers of **annotation** and **alignment**. In this chapter, we shall describe how we built such a corpus, which challenges we encountered and where we identified opportunities for improvement, in particular using triangulation approaches based on word alignment. These insights may serve as a blueprint for future works on other multilingual corpora.

We decided to use the **Europarl corpus** (Koehn 2005) as basis since it comprises all languages we are interested in, and its linguistic content, the transcribed debates of the European Parliament, is close to natural language use (Callegaro

Contributions

Many people contributed to different stages of our corpus construction building pipeline. Chiara Baffelli investigated how to optimize dependency parsing in Italian and Simon Clematide adapted her pipeline for French and Spanish. He also trained the parsing model for German. Mathias Müller worked on many parts of the annotation pipelines. The initial set of tokenization rules is the results of many fruitful discussions with Martin Volk. We use the example sets collected by him for English, French and German as unit tests for our tokenizer. Other examples have been provided by Chantal Amrhein, Mara Bertamini, Anne Göhring, Natalia Korchagina, Phillip Ströbel, Daniel Wüest and many others.

The design and implementation of our tokenizer and the definition of tokenization rule sets for all 16 languages that we deal with are the author's own work; similarly, the implementation of the processing pipelines and the design of the corpus database. Furthermore, the two algorithms described in Sections 3.2.1 and 3.2.2 have been implemented in SQL by the author.

2017). In fact, we frequently find colloquial expressions in the transcripts that we do not expect to find in corpora of formal written texts (e.g., "Mr President, I always thought being an MEP was a waste of time, but this really takes the biscuit here this evening."). Many corpora come from specific domains (see Chapter 2) and are, hence, less suited to study general linguistic usage. Some are designed to be as representative for linguistic usage as possible. Leech (1992) describes the goal of creating the British National Corpus (BCN), a comprehensive collection of British English text and speech material, as "to make it as far as possible representative of the full range of variation in the language", well aware that a corpus will never perfectly represent any language in its entireness.

In Europarl, we expect to find idiomaticity and colloquial expressions that, for instance, corpora from the legal domain (e.g., the JRC-Acquis corpus (R. Steinberger, Pouliquen et al. 2006)) typically do not possess. We also do know the **original language** of a speaker contribution in many cases, which allows us to issue queries depending on the translation direction. Last but not least, sentences in Europarl are comparatively short compared to other corpora. In our sentence segmented corpus, we count on average 26 tokens in English, 23 in German, 18 in Finnish and 28 in French. In contrast, in the JRC-Acquis corpus (ibid.), we count on average 45 tokens per sentence in English (headlines excluded).[1] Short sentences yield a better quality of statistical NLP tools that deal with whole sentences (e.g., parsers or word aligners).

The units that we get from our cleaned version of the Europarl corpus (named CoStEP; see Section 2.3.1) are aligned texts, each of which comprising the transcribed and in most cases translated speech of a member or a guest of the European Parliament. Written explanations of vote by one or more members, such as the one shown in Figure 3.1, are often appended to the oral comments given at the plenary sessions. These are found at the end of the comments and indicated as such in CoStEP. We refer to both oral and written contributions as (speaker) **turns**.

Below the level of turns, paragraphs are marked in Europarl and accordingly in CoStEP. The subdivision in paragraphs, however, is not consistent between languages in the original corpus data. We thus take paragraph boundaries as safe breaking points for **sentence segmentation**, identify other sentence boundaries within paragraphs during **tokenization** (Section 3.1) and re-join split sentences when performing sentence alignment in case there is no corresponding possible segmentation to be found in any other language.

On the basis of the resulting token sequences, we perform part-of-speech tagging and lemmatization in most languages (Section 3.2). We obtain **universal part-of-speech tags** (Petrov et al. 2012) by mapping the language-specific tagsets

[1] For comparison: The 'Oxford Guide to Plain English' (Cutts 2013) recommends "an average sentence length of 15–20 words."

> For us the report is a disappointment. It is of course good that we are studying how to create new jobs in the EU countries, but unfortunately there is no mention of the fact that we must also try to prevent the disappearance of jobs, particularly in the public sector. Unfortunately, the EU's policy, through the goal of economic and monetary union, means that many jobs are being cut in the public sector where many women work. The report entirely lacks an analysis in this area. We cannot agree with paragraphs 3, 4 and 24, which concern the necessity to coordinate economic policy at the EU level. We do not believe this is a way to create more jobs, mainly because, among other things, the industrial structures in the EU countries look very different. Of course there are also parts of the report which are positive, including the request to study unemployment among the young and to put forward proposals which the Member States can use to do something about youth unemployment. It is also important to have a switch in taxes so that tax on work is reduced while tax on energy and raw materials is increased, which is a policy the Green Parties are pursuing in the Member States. A reduction in working hours is also a good way to reduce unemployment.

Figure 3.1 – Example for a joint explanation of vote by three authors.

to the universal one. Using the tokens and their part-of-speech tags, we perform **syntactic dependency parsing** in our primary languages (English, French, German, Italian and Spanish) plus Swedish (Section 3.3).

We store tokens, structural information about sentences and texts, metadata from CoStEP, annotations and alignments (methods described in Chapter 4) in a **relational database**. Relational databases exhibit several features that prove beneficial both for modeling corpus data and for efficient retrieval of complex corpus queries. Only a few structural elements are required to represent text corpora in a **database schema** (Graën and Clematide 2015). We refer to the final corpus stored in a database as **database corpus** and to the database itself as **corpus database**.

One of the major challenges in automated corpus annotation based on statistical models is that those models have an inherent **error rate**. Especially word alignment, which, unlike sentence alignment or parsing, lacks an explicit definition and has evolved rather driven by statistical models than by imitating an existing linguistic structure, is error-prone. In order to reduce the impact of errors one can resort to only use the most frequent cases. This is what statistical machine translation does, but methods to select good corpus examples (Kilgarriff, Husák et al. 2008), for instance, also rely partially on frequency. In a similar vein, we present frequency-ranked lists of translation variants in Multilingwis (Section 5.2).

Another option to lower the risk of selecting an erroneous example by reason of such statistical errors is to **combine information from several layers**. The chance that all of them fail at the same point producing the same kind of error, thus yielding a false positive hit, is considerably smaller than an error made by a single source. We combine several layers of relations between tokens for our multilingual word alignment approach (Section 4.5) and for phraseme identification (Section 5.3).

The process of corpus preparation is not as straightforward as it reads in published works, which typically only reflect the last setup that was used to perform experiments on. We often encountered smaller or bigger issues several processing steps after they arose. A systematic tokenization error can, for instance, lead to wrong word alignment in particular cases that we only become aware of when analyzing word alignment statistics. We collected those errors and fixed them in the subsequent version of the corpus preparation pipelines. This backtracking approach led to several versions of our database corpus over several years.

We only keep those three versions that were used for data extraction at some point. All of them comprise the full list of speaker turns available in the respective CoStEP version they were built upon,[2] as opposed to corpora of much smaller size that we used for developing our processing pipelines. We will refer to these three 'full size Europarl' database corpora as FEP3, FEP6 and FEP9.[3] Table 3.1 shows the number of tokens for all languages included in each version in comparison to the number of tokens reported on the Europarl website[4] for the current Europarl release (v7) that we used for building CoStEP. We do not know how tokens have been counted in the Europarl corpus. When we process Europarl's raw files and remove all XML tags, we count on average 16 % more tokens than specified on the Europarl web page (Koehn 2012) (as 'words'); except for Polish, which apparently has been counted incorrectly in Europarl as the number of tokens cannot go up when the source material is shrunk.

When we describe the corpus preparation steps, we usually refer to the last corpus version, FEP9. Differences to the previous version are described to contrast different approaches and to motivate our decisions for making modifications. We started with our five primary languages (English, French, German, Italian and Spanish; see also Section 1.1) for FEP3. For FEP6, we added Finnish and Polish to have two more language families represented in our corpus. The last version, FEP9, comprises these seven plus nine more languages. From the 21 languages available

[2] FEP3 is derived from CoStEP version 0.9.0, FEP6 from version 0.9.4 and FEP9 from the final version 1.0. Systematic errors in CoStEP that we encountered while working on the respective corpus were fixed in subsequent versions of CoStEP.

[3] We maintain the original numbering to keep this document in line with envisaged release of the corpus data. Other version than these have not been used for data extraction.

[4] http://statmt.org/europarl/

Table 3.1 – Token counts in release v7 of Europarl (Koehn 2012) and three versions of our final database corpus. Languages marked bold are guaranteed to have translations for all turns. That is why there are fewer turns, and accordingly tokens, in FEP6, where we included Finnish translations as a requirement.

Language	Europarl	FEP3	FEP6	FEP9
Bulgarian				7 509 902
Czech	13 195 311			
Danish	47 761 381			
German	47 236 849	**41 119 084** −13 %	**38 085 206** −19 %	**41 107 021** −13 %
Greek				32 263 532
English	53 974 751	**43 176 169** −20 %	**39 952 337** −26 %	**43 151 584** −20 %
Spanish	54 806 927	**45 235 010** −17 %	**41 845 748** −24 %	**45 232 847** −17 %
Estonian	11 358 009			8 136 702 −28 %
Finnish	33 708 706		**28 453 158** −16 %	28 363 987 −16 %
French	54 202 850	**47 341 181** −13 %	**43 720 145** −19 %	**47 270 588** −13 %
Hungarian	12 606 986			
Italian	50 259 169	**42 652 193** −15 %	**39 458 151** −21 %	**42 648 100** −15 %
Lithuanian	11 512 131			
Latvian	12 085 228			
Dutch	53 487 257			42 954 617 −20 %
Polish	7 087 016		9 123 170 +29 %	9 334 433 +32 %
Portuguese	52 300 149			44 029 641 −16 %
Romanian	9 663 544			7 963 967 −18 %
Slovak	13 116 301			9 406 142 −28 %
Slovene	12 665 974			9 208 808 −27 %
Swedish	45 665 947			36 135 818 −21 %
Sum	596 694 486	219 523 637	240 637 915	454 717 689

in Europarl, we only excluded five, mostly due to the unavailability of a TreeTagger model. We made two exceptions: We used a different tagger for Swedish since we envisaged working with Swedish[5] and we left Greek untagged to see how well our methods (e.g., our multilingual word alignment approach described in Section 4.5) work on an – apart from tokenization – unprocessed language that uses a different script.

[5]In (Volk and Graën 2017), we use the data to explore the properties of multiword adverbs in English, German and Swedish.

All the described corpus annotation techniques and improvements together with word alignment information detailed in Chapter 4 form the basis for multilingual corpus queries. The implementation as database corpus enables to formulate and run virtually arbitrary queries efficiently, which we demonstrate by means of the applications that we present in Chapter 5. All the corpus processing steps performed here on the basis of the European Parliament's debates can likewise be applied to any other parallel or multiparallel corpus (see Chapter 2). We assume that the OpenSubtitles corpus would be particularly useful for investigating spoken natural language use.

Something that we did not address in corpus annotation process is the issue of code-switching, that is, a speaker changing her language temporarily. We frequently see single terms, mostly English ones, used in other languages, but also phrases or whole cited sentences.[6] If a foreign language is used by the original speaker, the translations of her speech predominantly also include that foreign expression (e.g., "General De Gaulle hat einmal von der paix des braves, vom Frieden der Tapferen, gesprochen." (original), "General de Gaulle once spoke of the paix des braves - the peace of the brave.", "el General de Gaulle habló una vez de la «paix des braves», la paz de los valientes.", "kenraali de Gaulle puhui kerran rohkeiden rauhasta (paix des braves)."). These unidentified chunks of other languages will be treated by the respective language models as if they belonged to the language in question and thus lead to (small) propagating errors (e.g., 'braves' in 'paix des braves' is tagged as an adjective and lemmatized to German 'brav' *'well-behaved'* like, for instance, 'braves Mädchen' *'well-behaved girl'*).

3.1 Tokenization

Initially, we did not dedicate much work to tokenization. The TreeTagger, which we decided to use for tagging and lemmatization (see Section 3.2), comes with a script that performs tokenization on the input stream before passing it to its proper tagging application. It was only later when we had processed all five primary languages and built tools for visualization that we encountered systematic problems that we could trace back to how we had tokenized the input texts.

We considered extending said tokenization script or enclosing it by pre- and post-tokenization in order to protect particular items that were erroneously split into two or more tokens and to split tokens that the script was systematically missing. In view of the future extension of our corpus to other languages and

[6] "Wenn er einen working-Tisch organisiert für democratisation and human rights, wunderschön!" *"If the Stability Pact were to organise a 'working' table conference for democratisation and human rights, all well and good."*

owed to several requirements that could not be easily integrated into the existing tokenization script, we implemented our own tokenizer with three features in mind: modularity, adaptability and traceability.

He and Kayaalp (2006) compare various tokenizers for the biomedical domain. They point out the need for standard tokenizers in order to ensure the interoperability of processing tools. Cruz Díaz and Maña López (2015) follow up with an analysis of more recent tokenizers, also for the biomedical domain. They observe disagreement to a large extent between the tokenization decisions of those tools for the test cases they had identified preliminarily. That observation is still in agreement with Habert et al. (1998), who concluded more than 15 years earlier: "At the moment, tokenizers represent black boxes, the behavior and rationale of which are not made clear."

Apart from rule-based tokenization, there exist statistical machine learning approaches to tokenization as well. For those approaches, a large amount of training material (i.e., original untokenized text with marked tokenization boundaries) is required. Jurish and Würzner (2013) argue that sufficient training material could be extracted from "treebanks or multi-lingual corpora". The problem with treebanks, however, is that they do not always comprise information about the original text as character sequence. When reconstructing the untokenized text from a list of tokens (as Jurish and Würzner (ibid.) had to resort to in one case), problematic cases may be overlooked and de-tokenized incorrectly. Beyond that, treebanks are available predominantly for standard text types of well-resourced languages. When it comes to non-standard texts (e.g., technical literature or historical text sources) or low-resourced languages, chances are that there is no or not sufficient material to learn a tokenization model from.

3.1.1 Cutter: Our Flexible Tokenizer for Many Languages

The question of what constitutes a word is a rather philosophical one. If we take, for instance, the dictionary entry 'time bomb',[7] do we deal with two words, 'time' and 'bomb', or do both parts together account for one word? If we opt for two words, what about the variants 'time-bomb' (24 % in FEP9) or 'timebomb' (6 % in FEP9)? However, if we define a word as everything that constitutes a common concept, we do not always have the ability to automatically decide where such a concept starts and where it ends since the individual parts of a concept like 'time bomb' may appear on their own or in other multiword expressions.

We can approach this division into concepts in a more elementary way. We leave aside for now the notion of words and go over to the notion of token. Tokens

[7] Compare, for instance, https://www.merriam-webster.com/dictionary/time%20bomb and https://www.macmillandictionary.com/dictionary/british/time-bomb.

are elements that we define to be our smallest units. They can be united to form phrases (see Section 4.4.1), but they cannot be split into smaller units.[8] In most cases, token boundaries are unambiguous; space characters and punctuation marks (less often the period) are typical characters that define beginning or end of a token.

A tokenizer that only splits a character sequence at space characters and before and after punctuation marks will already achieve a good accuracy, though missing all cases where the white space or punctuation marks belong to a token, along with those cases where we would want to split the character sequence in a different position. Tokens with white spaces are, for instance, numbers with five or more digits in some text (e.g., 50 000) or compounds consisting of a quoted expression and another noun.[9] Tokens containing periods are typically abbreviations (e.g., 'approx.', 'etc.', 'Ltd.') or ordinal numbers in some languages (e.g., German, Finnish, Polish).

To cover all these nontrivial cases, we systematically classified them and defined minimal test examples with their desired tokenization for each class, similar to units in unit testing. In so doing, we can guarantee that all cases are covered when none of these tests fails. Furthermore, extending the coverage of the tokenizer without invalidating previously correct tokenization decisions is possible by ensuring that all tests still pass after modification.

We make the underlying generic tokenization guidelines, our unit tests for all languages covered, the tokenization rules and the tokenizer itself available at http://pub.cl.uzh.ch/purl/cutter.

Implementation

Unlike other tokenization approaches that process text as a stream of characters from left to right, we identify tokens in a given text by means of regular expressions and 'cut them out'. That is why we named the tokenizer Cutter. Every time, a token has been cut out, we are left with a text part to the left of the identified token (potentially empty) and a part to the right (also potentially empty). We then continue processing both parts the same way we did for the entire text.

For that to work, the regular expressions that define our tokenization rules need to capture the whole input in various parts: identified tokens, surrounding

[8] "From a corpus-linguistic perspective, tokens also represent the minimal unit of investigation, the minimal character sequence that can be addressed in a corpus query." (Chiarcos, Ritz et al. 2012).

[9] They are numerous in the German part of our corpus: „Muttersprache plus zwei Fremdsprachen"-Formel, „Erhöhung der Subventionen an ineffiziente Landwirte"-Bericht, „European Masters of Excellence"-Aufbaustudiengänge, „Ich kann es nicht mehr ertragen"-Generation, „in der Europäischen Union hergestellt"-Etikett, …

white space characters and unprocessed text. Once a token has been identified, no further processing is applied to it. Surrounding white spaces can be retrieved as a special token class. This is particularly helpful if the original sequence of characters needs to be restored at a later time. By default, we omit the output of white space tokens, thus only returning linguistically relevant units. We proceed with the unprocessed text parts recursively until only empty parts remain. The resulting structure is a tree with all the leaves being tokens and the nodes corresponding to a particular rule that has been applied (see Figure 3.2).

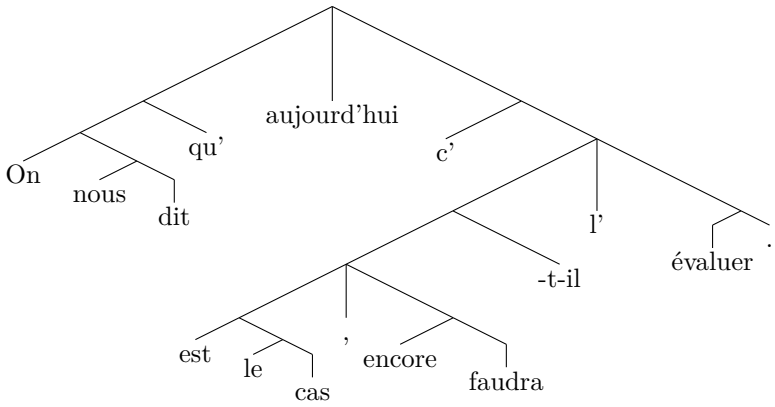


Figure 3.2 – The French sentence "On nous dit qu'aujourd'hui c'est le cas, encore faudra-t-il l'évaluer." as deconstructed by our tokenizer.

Tokenization rules can overlap, that is, the patterns of two different rules can both be applicable to a particular position of the input text. To determine the order in which rules are applied, we allocate each rule to one of several rule sets. There are two types of rule sets: language-specific and language-independent ones. The rule sets themselves are ordered such that more specific tokenization rules are applied before less specific ones and language-specific and language-independent rule sets are interwoven. Inside each rule set, the same principle is used for ordering the respective rules. In the end, we get an ordered list of regular expressions. When applied to input text, the first expression determines which character sequence is cut out.

In the example in Figure 3.2, the token [aujourd'hui][10] is treated first because it contains an apostrophe, and apostrophes are typically used in French to indicate elision of word-final vowel (as for [c'] and [l'] below). This would have let to the identification of a token [aujourd'] if the latter rule was applied first. The two remaining parts, left and right of [aujourd'hui], correspond to the left and right branches of the root node. On the left side, the next token to be identified by

[10]We use square brackets to denote token boundaries.

a (different) rule is [qu']. The remaining left side of that rule is then cut by the least specific and, hence, last rules. They identify the first character sequence terminated by a white space character ([On] and [nous]) and the remaining characters if there is no white space left ([dit]). On the right side of the root node, the first token to be identified is [c'], leaving only a right part for further processing. The same rule identifies [l'] in the next step. From there, [-t-il] is identified first, followed by [,]. The remaining parts on that branch are then processed with the default rules. On the rightmost branch, the rule for the sentence final period, which has precedence over the default rules, identifies the period and leaves only one token in one remaining branch.

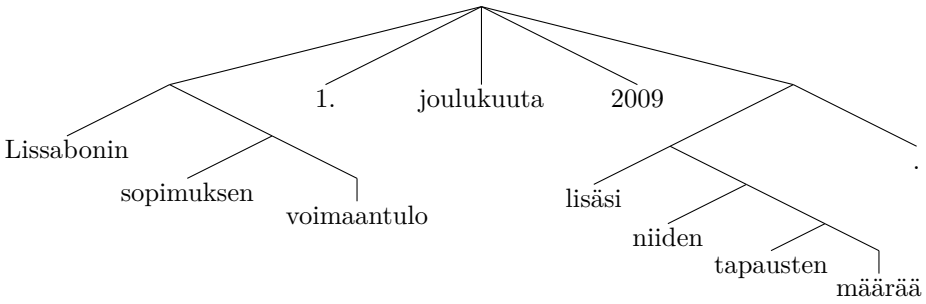


Figure 3.3 – The Finnish sentence "Lissabonin sopimuksen voimaantulo 1. joulukuuta 2009 lisäsi niiden tapausten määrää." comprises a date specification consisting of three tokens that have been identified by the first matching rule.

Common rules identify one token and leave a left and a right branch. A good example is the rule that identifies lexicalized words with an apostrophe in French (e.g., [aujourd'hui], [c'est-à-dire], [presqu'île]), which is applied first in Figure 3.2. It is, however, not necessary for a rule to follow this schema. In Figure 3.3, we see an example of a rule matching several tokens, namely parts of a date expression ([1.], [joulukuuta], [2009]). Date expressions follow a well-known scheme and appear frequently in different types of text (e.g., parliamentary debates, newspaper articles, diaries). By identifying these expressions at an early stage, we prevent other rules from matching parts of it.[11] This is the main idea behind our orderer rule sets: Whenever we can pinpoint a particular type of token or tokens (e.g., date expressions, XML elements, compound noun, number range), we cut it out and thus protect it against further examination. The more reliable the patterns of those tokens can be described and the more characters they extend over, the earlier we apply the corresponding rules.

[11] In a sentence final position, for instance, the year could be mistaken for an ordinal number if it is followed by a period (and the respective language expresses ordinal numbers with a period).

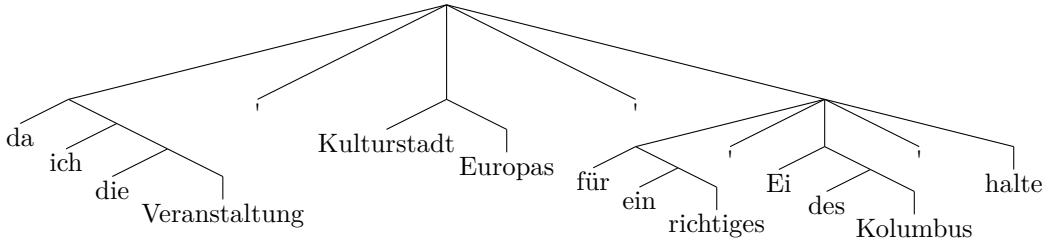


Figure 3.4 – This part of a German sentence “da ich die Veranstaltung 'Kulturstadt Europas' für ein richtiges 'Ei des Kolumbus' halte” has typewriter apostrophes instead of the proper left and right single quotation marks. The tokenization tree features inside branches for quoted parts.

Besides matching the remaining character sequences to the left and to the right of an identified token, we also permit other matching parts of a pattern to be marked as unprocessed so that they will be treated like the unprocessed parts to the left and to the right of what the pattern is matching. Figure 3.4 exemplifies a case that requires those ‘inside branches’. We often have to deal with text that uses the upright apostrophe (typewrite apostrophe) instead of enclosing single (typographic) quotation marks (likewise the ambiguous double typewriter quotation mark).[12]

Here, two cases conflict: the single quotation of a phrase as shown in Figure 3.4 and the word-final possessive apostrophe (in German used for words phonetically ending in /s/ (i.e., ‘s’, ‘z’ or ‘x’) and in English predominantly for words ending in ‘s’). In the example sentence shown in Figure 3.4, two of the four apostrophes follow directly an ‘s’ letter, which makes them candidates for forming a token together with the respective preceding continuous character sequences of non-space characters, that is, [Europas’] *‘of the Europes’* and [Kolumbus’] *‘of Columbus’*. For an algorithm to reject these options, grammar and world knowledge are required, in particular, that there is typically no plural of ‘Europa’[13] and that the determiner ‘des’ in ‘Ei des Kolumbus’ *‘egg of Columbus’* is already sufficient to mark the possession and thus the apostrophe would be unnecessary.[14]

[12]The absence or difficult accessibility of typographic quotation marks on computer keyboards certainly contributes to the widespread use of typewriter quotation marks and even if texts make use of the typographic quotation marks, those may be replaced by the typewriter ones to facilitate the processing with NLP tools (see Section 2.3.1).

[13]When speaking about Europe as an idea, one could, admittedly, have different ‘Europes’: “Gibt es heute nicht tatsächlich ein, zwei oder sogar drei verschiedene Europas?” *‘Are there not actually one, two or even three different Europes today?’* (found in “Die ZEIT (1946–2016)” at DWDS (W. Klein and Geyken 2010); `https://www.dwds.de/`)

[14]The inverse configuration “Kolumbus’ Ei”, which requires the apostrophe as genitive marker, is grammatically correct but in opposition to “Ei des Kolumbus” not idiomatic.

This distinction cannot easily be handled within the scope of a tokenizer. But we assume that a single apostrophe at the end of a non-space character sequence and followed by a space or punctuation mark by default constitutes a single token together with that character sequence. We also assume that a pair of two apostrophes is used to express some sort of quotation (main quotation in British English and subordinate quotation in several other languages; also used for highlighting of a term or expression). For this pair to match, we require that no word character is located to the left of the left apostrophe and to the right of the right one (typically a space). This is both times the case in Figure 3.4. In Finnish and Swedish, among other languages, the same quotation mark is used regularly for the start and end of a quotation, but the context of the quotation marks (word vs. non-word characters) can be used to disambiguate them.

Technical Details

We define our token patterns by means of Perl Compatible Regular Expressions (PCRE).[15] PCRE feature Unicode[16] character properties, named capturing subpatterns and subpattern assertions. Unicode character properties define classes of Unicode characters that share the same property, for instance, being an uppercase letter in any language (`\p{Lu}`), being a space character (`\p{Zs}`)[17] or belonging to the Cyrillic script (`\p{Cyrillic}`). Those classes can be negated, consequently matching anything that is not an uppercase letter (`\P{Lu}`), a space character (`\P{Zs}`) or does not belong to the Cyrillic script (`\P{Cyrillic}`). Classes can be combined and intersected.

Using those character properties in subpattern assertions, which in contrast to regular assertions can have a non-zero width, we set conditions on the context of a potential token that we want to identify. A prototypical token is delimited by spaces to the left and to the right. Another character class we typically find immediately preceding or following tokens are punctuation marks. Immediately preceding punctuation marks are commonly parentheses, quotation marks, hyphens (e.g., “EU-owned and/or -flagged”, “Herkunftsländer und -regionen”) or dashes (e.g., “–αυτό δε μπορούμε να το αγνοούμε–”, “–meiner Meinung nach–”, “–tesi che condivido–”). While the hyphens in these examples denote omission of a shared part (‘EU’ and ‘Herkunfts’) and, hence, should be recognized as part of the following character sequence, we want to split the other punctuation marks from the following sequence.

Apart from spaces and punctuation marks, the beginning or end of the entire character sequence is also a possibility that we want to permit as token boundary.

[15] (Hazel 1997); http://www.pcre.org/
[16] (The Unicode Consortium 2017); http://unicode.org/
[17] This class comprises 17 different space characters.

Since these are zero-width assertions, we cannot state our requisite as positive subpattern assertion over negated character properties (i.e., any character of the non-uppercase class, which includes everything but uppercase letters). Instead, we require the absence of positive character property (i.e., it must not be the case that there is an uppercase letter). This is done by means of negative lookbehind and negative lookahead assertions. They also cover the case that there actually is no letter at all.

Take the following Finnish sentence as an example: Raportissa ei mainita 12 000:tta Gazasta Israeliin ammuttua ohjusta, jotka uhkaavat vakavasti paikallista väestöä. *'The report makes no mention of the twelve thousand rockets fired on Israel from Gaza, which posed a very serious threat to the local population.'* We are interested in identifying tokens like [12 000:tta], numbers in one of the numerous Finnish cases, potentially having more than three digits and a space as thousands separator[18] to prevent subsequent tokenization from identifying the space (or the colon) as token boundary. For this purpose, we define the pattern for those numbers (digits intercepted by some space characters) followed by a colon and a sequence of lower case letters.[19] We prepend a negative lookbehind assertion and append a negative lookahead assertion to this pattern, both of a single letter class character.

Finally, we need to capture all parts for further processing. The tokenizer expects the leftmost and the rightmost capture to be further processed. All other capturing subpatterns are named, and are, depending on the identifier, likewise processed further (inside branches), filtered out (white spaces) or returned with their name as tag. The token identification rule for the described Finnish numbers is shown in Figure 3.5. The tokens identified by this rule will be tagged as `fiQnum`, which later serves to indicate which rule is responsible for which tokens and facilitates the following processing steps, for instance, tagging and sentence segmentation.

Advantages

Although token identification rules can become more complex than this one, for example, when they stretch across more than a single token or require a more specific context, complexity stays low due to the fact that every pattern (i.e., every token type) is defined on its own and the pattern scheme remains the same. Rule sets can be combined (e.g., to cope with code-switching) and both single rules

[18] This is the standard separator in Finnish. Other options are a comma (most English-speaking countries), a period (e.g., Austria, Germany, Italy, Spain) or apostrophes (Switzerland).

[19] For convenience, we do not make an effort to list all possible Finnish case endings for numbers.

```
/^(.*)(?<!\pL)
(?<fiQnum>\d+(?:\p{Zs}\d+?)+?:\p{Ll}+)
(?!\pL)(?<_>\s*?)(.*)$/uxUs
```

Figure 3.5 – Token identification rule for Finnish numbers with case endings as regular expression. The first line captures the left branch using a negative look-behind assertion for the immediate left context, the last line captures the right branch using a negative lookahead assertion for the immediate right context and additionally any potentially following white space character (named '_'). White space characters in the rule are ignored.

and rule sets can be deactivated separately. By only selecting basic rules, we can, for instance, limit tokenization to the detection of extra-linguistic elements such as XML tags, email addresses, URLs and file names.

If a particular case needs to be treated differently for a particular downstream application, a rule can also be replaced. In the example in Figure 3.2, we tokenize the French sentence as "… [encore] [faudra][-t-il] [l'][évaluer][.]". Alternatively, we could want the euphonic 't' ('t' euphonique) in 'faudra-t-il' to be separated from the enclitic 'il', so that it is analyzed like in "[fait][-il]", "[doit][-il]" or "[s'][agit][-il]". In that case, we replace the rule that identifies French enclitics (including possible euphonic 't's) with a variant that uses an extra capturing subpattern for '-t', which will analyze the French from Figure 3.2 as "… [encore] [faudra][-t][-il] [l'][évaluer][.]". We could also decide to drop the euphonic 't' altogether by making the subpattern non-capturing.[20]

A behavior we see in many tokenizers is that they, expectedly, treat unknown character sequences according to their rules (if rule-based) or statistical models (if trained on textual data with annotated tokenization boundaries). To make specific token types (e.g., XML tags, URLs or text-specific units such as 'A5-0210/2001', which identifies a report targeting the welfare of pigs) pass unscathed through the tokenizer, we either need to replace it by some unambiguous character sequence beforehand and restore the original sequence afterwards, or we use existing patterns that the tokenizer in question handles well (e.g., XML tags) to store the unknown token type as payload and re-extract it after it has passed the tokenization encapsulated. In any case, we need to identify those types before tokenization and restore them afterwards.

With Cutter, we have two options to meet this challenge: We either write rules for each specific type and include them in the rule sets, or we mark them as

[20]We do not recommend dropping anything apart from white spaces. In our view, the role of a tokenizer is to identify token boundaries. It may split contractions such as "won't" into '[wo]' and '[n't]', but restoration of the original words 'will' and 'not' should be done in a post-processing step if required in this form by downstream applications.

protected beforehand. The latter is achieved in Cutter with the aid of two special Unicode code points, U+2402 and U+2403 from the 'Control Picture' block, which are graphical representation of the control characters 'start of text' and 'end of text', respectively. Code points in the Control Picture block should only appear in texts that describe control characters. Any character sequence enclosed by these two code points is identified by the very first rule as a token.

We take advantage of this feature to handle abbreviations. Language-specific abbreviation lists are matched against any character sequence ending with a period[21] in the text to undergo tokenization and every hit is marked with the aforementioned code points. Those abbreviations that consist of a character sequence that exists as a lexical unit represent a peculiarity. We can, for instance, not allow the German abbreviations 'Abt.' for Abteilung *'division, section'* or 'Art.' for Artikel *'article, item'* in the abbreviation list since Abt *'abbot'* and Art *'kind'/'type'* are valid German words, and we would otherwise always identify them as abbreviations in a (declarative) sentence final position. Instead, we have to handle those cases with rules that take more context into account.

Limitations

In the case of 'Art.', we know that it is typically followed by a number (e.g., "dass … öffentliche Debatten nach [Art.] [8] der Geschäftsordnung des Rates stattfinden müssen" *'public debates be held in accordance with Rule 8 of the Council's Rules of Procedure'*). We can, therefore, implement a rule that identifies 'Art.' as token if followed by a number. Unfortunately, that would also be applied in cases like "Das ist nicht die erste Katastrophe dieser [Art][.] [1994] wurden in der Region Tindouf 30 000 Menschen obdachlos …" *'It is not the first such calamity. In 1994, 30 000 people were made homeless in the Tindouf area …'*. A further refinement of this rule to smaller numbers only (no year dates) would solve the issue for these two examples, but they indicate the general problem that in some cases, the determination of the right token boundaries can hardly be concluded from the surface form alone.

Possible clues regarding the correct tokenization in such difficult cases come from morphology ('Abt' is a masculine and 'Abteilung' a feminine noun), phraseology ('zweit.' as abbreviation of 'zweiteilig' *'bipartite'* is typically not preceded by 'zu', the expression 'zu zweit' *'in pairs'* typically is), syntax (we cannot have two uncoordinated finite verbs in the same clause) or semantics (the decision whether

[21] Abbreviations do not necessarily terminate with a period. Cases with no particular abbreviation symbol typically do not require any special treatment (e.g., acronyms, units of measure). When a symbol other than the period is used, it depends on whether that symbol is ambiguous and could, unrecognized, lead to wrong tokenization. The word-central colon to mark omission of a character sequence in Swedish (e.g., 'ö:a' for östra *'eastern'*) is an unproblematic case.

a character sequence in question is an abbreviation or a regular word plus a period cannot be fixed onto any other feature and only becomes clear taking into account the meaning of the surrounding text).[22]

In these undecidable cases, it is thus an option to make a decision for one or the other tokenization, continue processing the tokenized text with downstream applications and, in case of error or improbable results, backtrack and revert to the alternative tokenization. Also, probabilistic tokenizers such as the one described in (Jurish and Würzner 2013) may be able to handle ambiguous cases like these by means of an abstract feature representation. However, a large amount of training material is required as they occur infrequently.

Furthermore, we have to provide those examples in German since, in our judgment, most other languages show fewer ambiguities regarding token boundaries. This is partially due to its peculiarity to capitalize all nouns, including nominalized word of virtually any part of speech, which, given that sentences typically start with an uppercase letter, brings forth more potential sentence boundaries.

Sentence Segmentation

New-line characters in the character stream provided to Cutter are regarded as definite sentence segment boundaries and processed separately. Rules that identify the end of a sentence return a zero-width token tagged as end-of-sentence marker. These markers belong to the same class of tokens that we use for white spaces, which is usually not returned unless explicitly requested.

When we have to decide whether a number followed by a period is an ordinal number (for those languages that use a period to express them) or a cardinal number at the end of a sentence, we look ahead at what follows the period. If we find a capitalized function word (e.g., a discourse marker), it presumably is capitalized on account of starting a new sentence, and we thus can issue an end-of-sentence marker between the period and the following character sequence.[23]

[22] Take, for instance, the proper name 'Franz' and the adjective 'französisch' *'french'*, which can be abbreviated to 'franz.' and needs to be capitalized in some combinations: "200 Jahre [Franz.] Revolution in Deutschland", "Mitglied der [Franz.] Akademie", "Personifikation der [Franz.] Republik" or "Die hiervon betroffenen Überseegebiete sind: [Franz.] Westafrika, [Franz.] Äquatorial-Afrika, St. Pierre et Miquelon, die Kommoren, Madagascar und abhängige Gebiete, [Franz.] Somaliland, Neukaledonien und abhängige Gebiete, [Franz.] Ozeanien, die südlichen und antarktischen Gebiete, die autonome Republik Togo, das franz. Treuhandschaftsgebiet Kamerun, Belgisch-Kongo und Ruanda-Urundi, Italienisch Somaliland, Niederländisch Neuguinea.". In contrast, when 'Franz' is first or last name of a person, we need to identify the period as separate token: "Kein schlechtes Wort, meint strahlend ZEW-Chef Wolfgang [Franz][.] Minderwertigkeitskomplexe haben Franz und seine 130 Mitarbeiter schon lange nicht mehr." (all examples found in DWDS corpora)

[23] The position of the empty end-of-sentence token is marked in the following examples: "Jedenfalls bis [50][.][] Dass alte Männer dick sind, ist ja normal."; "Ich glaube, das war Nr. 22 und

We also issue an end-of-sentence marker following any remaining punctuation mark unless it is itself followed by closing quotation mark, in which case the marker is issued after that one. If a declarative sentence ends with an abbreviation that is denoted by a period (e.g., 'etc.'), the sentence-final period is typically omitted in all languages we deal with to avoid reduplication. This makes it challenging to correctly identify sentence boundaries in such cases. Here, we can again revert to typical sentence-initial tokens (capitalized function words) and combine those with typical sentence-final abbreviations ('etc.' and 'et al.' vs. 'i.e.' and 'Mr.').

3.2 Part-of-speech Tagging and Lemmatization

We use the *TreeTagger* (Schmid 1994, 1995) for tagging and lemmatization (i.e., assigning the respective tokens their corresponding base forms) for all languages except Greek, which we do not process further after tokenization, Italian, which we process with *Trigrams'n'Tags (TnT)* (T. Brants 2000) as Baffelli (2016) found it to perform better on Italian than the TreeTagger with its pre-trained language model, and Swedish, where we use *Stagger* (Östling 2012, 2013). For German, we apply both the TreeTagger and the *Clevertagger* (Sennrich, Volk and Schneider 2013), but continue processing with the tags generated by the Clevertagger, as we determined by manual inspection that the latter handles cases such as particle verb prefixes and multiword adverbs (see below) better than the former. Since the Clevertagger, unlike the other taggers, does not assign lemmas to tokens, we revert to the TreeTagger's analysis for German lemmas.

Taggers and Resources

All four tools are statistical taggers, which need to be trained on correctly tagged (and lemmatized) data. In Table 3.2, we list the resources on which the respective statistical models have been built.

The TreeTagger learns *decision trees* from training data. It builds on trigrams, that is, in the resulting model, the tags of two preceding tokens are used to determine tag probabilities of the token in question. In addition, it uses the training data for learning tag probabilities attached to suffixes,[24] also represented as tree structures, from the training corpus such that the longest informative suffixes starting with the respective last characters below the root node are represented as paths from the root to a leaf node.

[25][.][] Dafür war ich Euch sehr dankbar."; "Arme kreisförmig hinten nach unten schwingen bis zur Haltung der Abb. [29][.][] Hierbei ausatmen." (all examples found in DWDS corpora)

[24]Note that these are not grammatical suffixes.

Table 3.2 – Taggers, training corpora and tagsets. The last column reflects the number of tags we observe in our tagged corpus; the respective tagsets may comprise additional tags that have not been assigned to any token.

Language	Tagger	Resources (training corpus, tagset, …)	Tags
Bulgarian	TreeTagger	Bulgarian Treebank; BulTreeBank Morphosyntactic tagset (Simov et al. 2004)	515
Dutch	TreeTagger	*training corpus unknown*	42
English	TreeTagger	Penn Treebank (Marcus et al. 1993); Penn Treebank tagset (Santorini 1990)	44
Estonian	TreeTagger	Corpus of morphologically disambiguated Estonian texts (Habicht et al. 2000)	365
Finnish	TreeTagger	FinnTreeBank (Voutilainen et al. 2012); morphological tagset reduced to the first three features (personal communication)	591
French	TreeTagger	*training corpus unknown*	33
German	Clevertagger	TüBa-D/Z (Telljohann, Hinrichs and Kübler 2004); SMOR (Schmid et al. 2004); Stuttgart-Tübingen Tagset (STTS) (Schiller et al. 1995)	54
	TreeTagger	*training corpus unknown*; Stuttgart-Tübingen Tagset (STTS) (Schiller et al. 1995)	53
Italian	TnT	Italian Stanford Dependency Treebank (Bosco et al. 2014) converted to Universal Dependencies (see also Baffelli 2016)	41
Polish	TreeTagger	National Corpus of Polish (Przepiórkowski et al. 2008); morphological tagset (Patejuk and Przepiórkowski 2010)	803
Portuguese	TreeTagger	Bosque 8.0 (see Gamallo and Garcia 2013); simplified morphological tagset (Garcia and Gamallo 2010)	72
Romanian	TreeTagger	MULTEXT-East ”1984” annotated corpus 4.0 (Erjavec, Barbu et al. 2010); morphological tagset	432
Slovak	TreeTagger	Slovak National Corpus (Horák et al. 2004); simplified morphological tagset	69
Slovene	TreeTagger	ssj500k corpus (Krek et al. 2015); morphological tagset (Erjavec, Fišer et al. 2010)	1223
Spanish	TreeTagger	Spanish CRATER corpus (McEnery et al. 1997); EAGLES- conformant tagset (Leech and Wilson 1994)	69
Swedish	Stagger	Stockholm-Umeå Corpus (SUC) (Gustafson-Capková and Hartmann 2006); SALDO morphological lexicon (Borin and Forsberg 2009)	26

Suffix trees are particularly beneficial with regard to part-of-speech tagging for languages that avail themselves of derivational suffixes, which includes most European languages.[25] The part of speech of a token is, hence, often evident by the suffix, that part that stemmers (e.g, Porter 1980) remove to obtain the part of a word that bears the meaning: its stem. The documentation of Snowball (Porter 2001), a programming language for defining stemming algorithms, states that Indo-European and Uralic languages are both "amenable to stemming".

When tagging with the TreeTagger, we exploit two useful features: pre-tagging and external lexica. Pre-tagging means to externally provide tag probabilities per token to the tagger. We make use of this feature to predetermine the tags (i.e., only providing one tag with a probability of 100 %) of tokens that received particular tokenization tags such as numbers (including those with white spaces), URLs or quotation marks.[26] We also pre-tag frequent multiword adverbs as they tend to get tagged incorrectly (see Volk, Clematide et al. 2016).

Similar to pre-tagging, the second feature, external lexica, allows us to equip the tagger with definition of word forms and their tags. In contrast to pre-tagging, we may only list the options here, without specifying their probabilities. An external lexicon, however, allows for an optional lemma for each word form, which is particularly useful for providing lemmas for compound nouns in German (e.g., 'Menschenrechtsklauseln' *'human rights clauses'*) as the training algorithm will not have seen many of them due to their productiveness.[27] We manually added frequent compound nouns (e.g., 'Schattenberichterstatter', 'Verhandlungsrichtlinien') and other parts of speech that the part-of-speech tagger missed in the first run (e.g., 'interinstitutionelle', 'nachgewiesenermaßen', 'gegenzusteuern') together with their respective lemmas to the German external lexicon.

Since we know the members of the European Parliament (see Section 2.3.1), we also appended their names to the external lexicon to ensure that they are recognized as proper nouns. We also include capitalized word forms that appear unchanged in most other languages as proper nouns. In case a proper noun coincides with a common noun, both corresponding tags are included as options.

Trigrams'n'Tags (TnT) implements a second-order Markov model, that is, a Markov model that determines the tag for the token in question based on at most

[25] Clackson (2007) describes Proto-Indo-European morphology. The concept of part-of-speech tagging presupposes that each token bears one identifiable grammatical category, which is typically true for inflected or isolating languages, but "for agglutinative or even polysynthetic languages, it is beneficial to use smaller units than word forms as the basis for statistical processing, in order to avoid data sparseness" (Rios 2015).

[26] If the tagger model has been trained on a corpus with only typewriter quotation marks, it will not recognize typographical one. The same is true the other way round.

[27] "A major problem in statistical POS tagging for German is the complex morphology of German, which results in many inflected or compounded forms which have never been observed during training." (Sennrich, Volk and Schneider 2013)

the previous two tags assigned. It also estimates lexical and suffix probabilities from the training data. The latter is used to predict the part of speech for unknown words, that is, word forms that have not been observed during training.[28]

Stagger uses a model that employs a set of features to predict the tag. Its "feature set takes into account the previous tag and previous pairs of tags in the history, as well as the word being tagged, spelling features of the words being tagged, and various features of the words surrounding the word being tagged." (Collins 2002) and thus does not differ much from the previously described approaches with regard to the context looked at when determining the tag for a given token. In addition, the tagger uses *word embeddings* (Collobert and Weston 2008) to constitute a feature that models how well the word form of the token in question harmonizes with its left and right context.[29] The morphological lexicon *SALDO* (Borin and Forsberg 2009) is used to confine the search space for the open (i.e., productive) word classes to those tags that are found in the lexicon.

The Clevertagger builds upon *conditional random fields* (Lafferty et al. 2001), which is a method for learning models to label sequences. As features to predict the tag sequence from, a window of five tokens centered at the token in question, the last two assigned tags and several features on the token level are used. One of them is a list of possible part-of-speech tags computed by morphological analysis. As resources for that analysis, two different system are used: the finite-state morphological analyzers *SMOR* (Schmid et al. 2004) and *Morphisto* (Zielinski et al. 2009). In case a word form has not been observed in the training data, morphological analysis limits the set of possible tags for the token in question and thus increases overall accuracy, in particular when applied to text types different from the training material.

Schmid (1994), T. Brants (2000), Östling (2012, 2013) and Sennrich, Volk and Schneider (2013) report an accuracy of above 95 %, that is, an erroneous tag is assigned to less than every 20th token in the input token sequences.

Tagsets

Language-specific tagsets show a wide variation in Table 3.2. The tagset used in the Stockholm-Umeå Corpus (SUC) (Gustafson-Capková and Hartmann 2006) only differentiates between 26 basic part of speech,[30] while morphological tagsets encode all possible combinations of morphological features and thus easily account

[28] Note that these are, equal to suffixes calculated by the TreeTagger, merely word-ending character sequences not related to grammatical suffixes.

[29] The effect of these word embeddings is reported to minimally increase tagging accuracy.

[30] 22 grammatical categories, three delimiters (i.e., brackets and punctuation) and one (undocumented) tag assigned primarily to URLs.

for several hundred tags. The tagset used in the Slovene JOS[31] corpora (Erjavec, Fišer et al. 2010) consists of 1902 of such feature combinations, 679 of which never got assigned to any of the more than 9 million tokens in our corpus.[32]

Tags in a tagsets can thus distinguish small morphological differences such as the aspect of Slavic verbs, more general differences such as number (singular or plural) or only denominate the principal grammatical category. Not all features are equally probable to be detected correctly. Džeroski et al. (2000) investigating the effect of different tagsets on tagging Slovene texts found that "inflectional features are much harder to predict than lexeme ones". An argument in favor of a smaller tagset is that, given a complex morphological tagset, the probability of error by getting all but one feature right is higher, while it is harder to select the wrong tag given fewer tag categories. Correctly recognized fine-grained tags, on the other hand, support a statistical tagger in making the right decision for neighboring tokens (ibid.).

Déjean (2000) is concerned with the question of what kinds of tagsets are beneficial to syntactic parsing. His statement that "the quality of a tagset does not depend on the quantity of tags" is to be seen with regard to this purpose. However, if parsing is the objective of tagging, several tags may be treated uniformly in parsing so that tagging errors only concerning a minor grammatical subcategory that is of little or no importance to syntax make no impact in the end.

We map every language-specific tagset to the so-called universal one (Petrov et al. 2012), more specifically, to the first proposed version consisting of 12 different tags. Mappings for common tagsets are available online;[33] in case of morphological tagsets, the correspondence is typically determined by the main category of a tag.

There are several advantages of having a reduced part-of-speech tagset on the one hand, and having one that is shared among all languages in a parallel corpus on the other hand. First of all, it facilitates our work with the corpus that we issue queries with part-of-speech restrictions without having to use a language-specific tag or an expression for referring to a set of tags. These queries using universal tags can easily be transferred from one language to another by just changing the language identifier. Moreover, results can be compared quantitatively with regard to part of speech.

Applying the mapping to universal part-of-speech tags to all individual tagsets, we get the tag distribution shown in Figure 3.6. Two of the universal categories are not available in all languages: Bulgarian, Estonian, Finnish, Polish, Slovak and Slovene do not exhibit determiners; Estonian, Finnish, French, Italian, Polish and Portuguese do not exhibit particles.

[31] 'Jezikoslovno označevanje slovenščine' *'Linguistic Annotation of Slovene'*

[32] The tagset is documented online at `http://nl.ijs.si/jos/msd/html-en/`.

[33] `https://github.com/slavpetrov/universal-pos-tags`

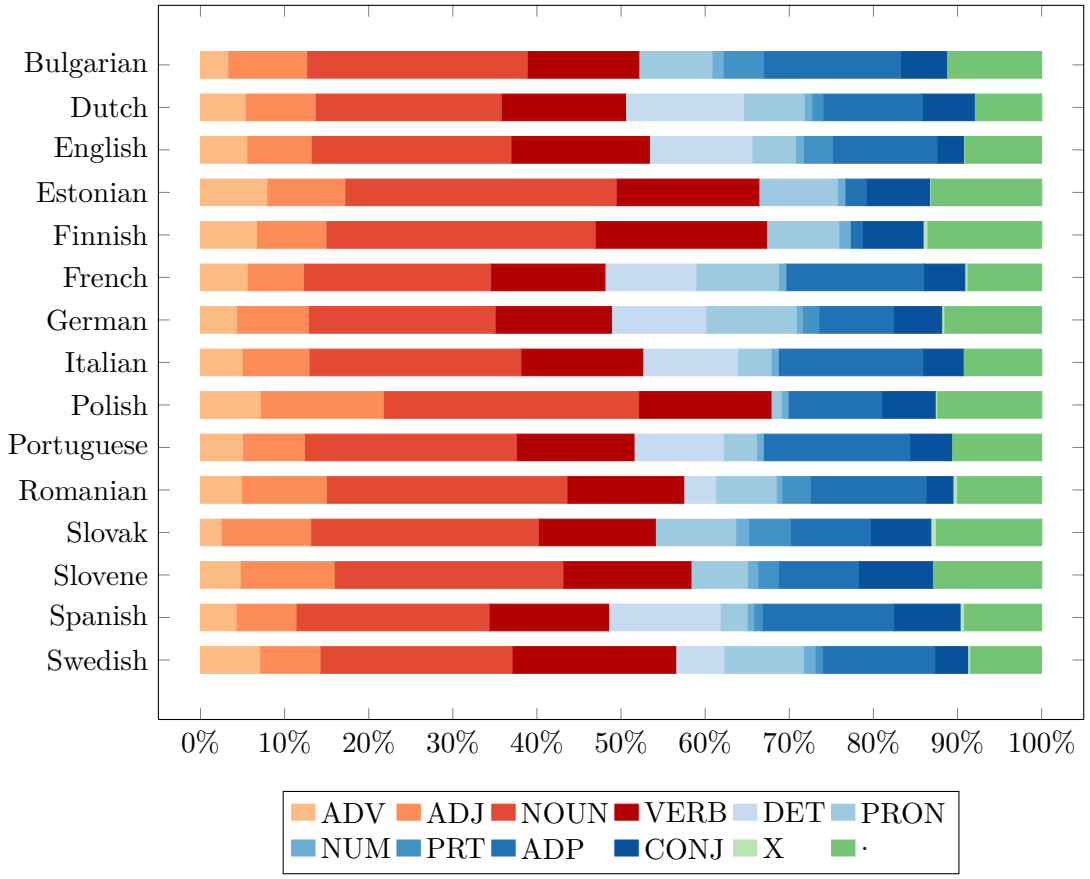


Figure 3.6 – Distribution of universal part-of-speech tags (Petrov et al. 2012) among the languages that we tagged. The absence of particular part-of-speech tags (e.g., determiner (DET) in Estonian or Finnish) is either because the language does not command that part of speech or the language-specific tagset has no equivalent tag (e.g., particles (PRT) in French or Portuguese).

Slavic languages do not feature determiners, apart from Bulgarian. In Bulgarian, however, determiners are realized as enclitics (like most articles in Romanian and also definite articles in Swedish). The task of a tagger is to attach exactly one tag to each token. The noun – being a content word, that is, belonging to an open word class and having determiners depending on it – is considered the more important property of a token consisting of both noun and determiner and, hence, the token is tagged as noun. Finno-Ugric languages neither express definiteness with individual tokens.

As regards particles, their use is envisaged for confined cases in French, Italian and Portuguese in the definition of universal part-of-speech tags.[34] For several reasons they are not present in our corpus: In some cases, there is no correspondent tag in the language-specific tagset so that we cannot map it accordingly. We also use a different tokenization in French for the euphonic 't' (see previous section), which is why we cannot assign it a tag separately. Finally, in Italian, the particle tag is only used for English possessive markers, but we do not separate them from the noun that they are attached to because often proper names are concerned (e.g., Lloyd's, Fisherman's, Sotheby's), which we prefer to keep unsplit and tag as nouns.

Lemmatization

Some taggers optionally predict lemmas in addition to tags as a byproduct of tagging. During training, a tagger observes numerous word forms and tags. If the training corpus also comprises lemmas, the tagger can learn the applicable lemmas given a particular word form and the tag assigned to each respective token. The TreeTagger's strategy is to output every lemma that has been observed during training with the chosen tag independent of their frequency.[35]

In German, nominalized verbs and declined nouns (often dative plural) frequently coincide in their forms (e.g., 'Pflanzen' *'plants'* (all cases plural)/*'planting'* (derived noun), 'Mitteln' *'averages'* (dative plural)/*'averaging'* (derived noun)), but also lexically unrelated words (e.g., 'Westen' *'vests'* (all cases plural)/*'west'*). In some cases, we find three alternative lemmas (e.g., 'Schmieden' *'smiths'* (dative plural)/*'smithy'* (nominative singular)/*'forging'* (derived noun)). In these cases, we can look in the translations for evidence to help disambiguate the lemmas. For that purpose, we exploit lemmatization in other languages via word alignment (see Section 4.4). Our approach is detailed in the next section.

Trigrams'n'Tags and the Clevertagger do not provide for lemmatization. For German, we revert to the lemmas issued by the TreeTagger; for Italian, we use the morphological analyzer Morfette (Chrupała et al. 2008), which jointly assigns morphological tags and lemmas to a token sequence. The Italian Stanford Dependency Treebank (Bosco et al. 2014) was used to train Morfette's statistical model for Italian (Baffelli 2016).

Stagger generates output in the CoNLL-X format, which includes besides the word form from the input, lemma, part-of-speech tags[36] and sets of morphological

[34] http://universaldependencies.org/docsv1/

[35] The Bulgarian tagging model has been trained without lemmas.

[36] Two different kinds of part-of-speech tags are allotted in this format: coarse-grained and fine-grained ones. However, Stagger does not make a difference between them.

tags. The way Stagger deduces the lemma is undocumented, nevertheless it is clear from the code that there is always an unambiguous lemma being returned.

3.2.1 Interlingual Lemma Disambiguation

We exploit word alignment (see Section 4.4) to disambiguate the aforementioned ambiguous lemmas. Our approach is similar to the one described in (Volk, Amrhein et al. 2016, Section 5), but uses evidence from all aligned languages instead of just one. We perform the following steps for lemma disambiguation:

1. We first calculate the global distribution matrix of lemma correspondences based on optimal alignments ($D_a : \Lambda \times \Lambda$ with Λ being the entire set of lemmas in all languages). Optimal alignments are those, that are supported by all four word aligners (see Section 4.4.1) in both directions where applicable:

$$D_a(\lambda_s, \lambda_t) = p_a(\lambda_t|\lambda_s) \tag{3.1}$$

   The lemma alignment distribution is calculated based on the frequencies f_a such that the alignment probabilities p_a of all target lemmas $\lambda_{t'}$ given a particular source lemma λ_s sum up to 1:

$$p_a(\lambda_t|\lambda_s) = \frac{f_a(\lambda_s, \lambda_t)}{\sum_{\lambda_{t'}} f_a(\lambda_s, \lambda_{t'})} \tag{3.2}$$

2. Each lemma alternative λ_s^i of a given ambiguous lemma is looked up in the same language part of the corpus. If one of the lemmas has not been seen in any other context, it disqualifies for disambiguation as we will not be able to calculate its probability. If only one lemma option is left by disqualification of the other options, it is selected as the correct lemma without further treatment.

3. Similar to the lemma distribution matrix, we only use optimal alignments to select corresponding tokens for disambiguation. From each of these tokens, we look up the lemma alignment probability between its assigned lemma λ_t (if appropriate) and each of the lemma alternatives: $p_a(\lambda_s^i|\lambda_t)$

4. The lemma alternative with the highest overall probability is chosen as replacement of the set of lemma alternatives:

$$\lambda_s^{final} = \underset{\lambda_s^i}{argmax} \sum_n p_a(\lambda_s^i|\lambda_t^n) \tag{3.3}$$

If we divided the probability sum by the number of optimally aligned tokens with lemmas, we would get an average lemma correspondence probability for each of the lemma alternatives. Since we are only interested in the one that yields the highest overall probability, we pass on the normalization.

Table 3.3 – Parallel example sentences with ambiguous lemma in German ('gehören' *'to belong to'* or 'hören' *'to hear'*). The underlined tokens in the other languages show optimal alignment with the ambiguous German lemma.

Language	Sentence	Lemma
Dutch	Ik heb gehoord dat er boeren zijn die in plaats van te melden dat zij een dier hebben met BSE, dat dier liever doodschieten en begraven.	horen
English	I have heard that there are some farmers who, rather than report they have an animal with BSE, shoot that animal and bury it.	hear
Finnish	Olen kuullut, että jotkut maanviljelijät mieluummin ampuvat ja hautaavat BSE:tä sairastavan eläimen kuin tekevät siitä ilmoituksen.	kuulla
French	J'ai entendu dire que certains éleveurs, plutôt que de rapporter un cas d'ESB lorsqu'il se présente, abattent l'animal et l'enterrent.	entendre
German	Ich habe gehört, dass manche Landwirte ein an BSE erkranktes Rind lieber erschießen und vergraben, als den Fall zu melden.	gehören ∨ hören
Spanish	Tengo entendido que hay algunos ganaderos que, en lugar de informar de que tienen un animal con EEB, matan al animal y lo entierran.	entender
Swedish	Jag har hört att det är en del jordbrukare som i stället för att rapportera att de har ett djur med BSE, skjuter det djuret och begraver det.	höra

A sample sentence with ambiguous German lemma is shown in Table 3.3. Here, the token with the word form 'gehört' could not be assigned an unambiguous lemma as 'gehört' is both the past participle of 'gehören' *'to belong to'* and 'hören' *'to hear'*. The bilingual word alignment yields optimal alignments for tokens in six other languages.[37] We show the lemma alignment probabilities for both lemma alternatives given the respective aligned lemma in all other languages in

[37] Three other languages show suboptimal word alignment for that source token.

Table 3.4. The sum of alignment probabilities for the lemma ‘hören’ is higher than for ‘gehören’, which makes our algorithm select ‘hören’ as the correct lemma and discard ‘gehören’.

Table 3.4 – Lemma alignment probabilities for the sample sentences in Table 3.3. The sum of probabilities is higher for ‘hören’ than ‘gehören’.

		$p_a(\lambda_s\|\lambda_t)$	
Language	λ_t	λ_s = gehören	λ_s = hören
Dutch	horen	0.2339	0.2909
English	hear	0.1515	0.3782
Finnish	kuulla	0.1143	0.3192
French	entendre	0.0694	0.1482
Spanish	entender	0.0043	0.0213
Swedish	höra	0.3314	0.2779
	$\sum p_a$	0.9047	1.4357

Our approach does not take into account that the alternatives may coincide in related languages. Dutch, for instance, shows a similar alignment probability for both alternatives, since ‘horen’ *‘to hear’* and ‘horen bij’ *‘to belong’* both comprise the lemma ‘horen’.[38] The alternative ‘behoren’ *‘to belong’* has approximately the same probability as ‘horen’ given the German lemma. In Swedish, on the other hand, the compound verb ‘höra till’ *‘to belong’* is considerably more frequent (almost four times) than the single verb ‘tillhöra’, which results in the contrasting numbers in Figure 3.4. Knowing this similarity between members of the Germanic language family, a future improvement of this algorithm could investigate the effect of introducing language-specific or language-family-specific weights into Equation 3.3.

In contrast to FEP6, where we used the same approach but did not disambiguate lemmas of function words, we performed disambiguation in FEP9 for all ambiguous lemmas. We leave the ambiguous set of lemmas untouched only in cases where none of the lemma alternatives can be found in the corpus, no optimal alignment exists or none of these alignments holds a lemma.

Applying our approach to the whole FEP9 corpus, we see most disambiguated lemmas in Finnish and German (approximately 200 000 each). Finnish lemmas include a symbol for morpheme boundaries and the ambiguity frequently only consists in a different position of this boundary, which renders German the lan-

[38] The preposition ‘bij’ does not prevent the alignment between the verbs to be optimal.

guage with most ambiguities in terms of TreeTagger models. In the Dutch part, we count approximately 190 000 cases, in Slovak 115 000, in Estonian 91 000, in Slovene 12 000 and in Romanian 4000.

Ambiguous lemmas in Italian (69 cases) and English (33 cases) only consists of erroneous cases. In Italian, numbers (e.g., 42, 1.2, 43 000), number ranges (e.g., 1996–2004), acronyms (e.g., JEREMY) and some rare words (e.g., X-ray) are assigned their word form plus an alternative lemma, which is always 'sconfitto' *'conquered'*. In English, some lemmas are provided with an alternative in capitalization, which suggests inconsequent lemmatization in the training data.

Evaluation

Since most real lemma ambiguities are found in German, disregarding the alternative morpheme boundaries in Finnish, we evaluate our approach on German. For this purpose, we randomly select 100 German sentences comprising at least one ambiguous lemma that has successfully been processed by our approach. In total, we count 114 disambiguated lemmas in the 100 sentences. The majority (61 cases) is concerned with the ambiguity between the personal pronouns 'sie' *'she'*/*'they'* and 'Sie' *'you'* (polite form of address). The latter is typically written with initial capitalization, but the former also requires capitalization in sentence-initial position.[39] All other cases with capitalized 'Sie', including other grammatical cases: 'Ihr' (genitive) and 'Ihnen' (dative), can be expected to be the polite form.

Unfortunately, the capitalized 'Sie' does not exist as single lemma in our corpus. According to the TreeTagger's training method to remember tag and lemma combinations for a given word form, the sentence-initial pronoun 'Sie' with lemma 'sie' and occurrences of pronoun 'Sie' with lemma 'Sie' lead to a collapsed entry that assigns each occurrence of pronoun 'Sie' both lemmas. The same applies to other grammatical cases. The lowercased pronoun 'sie', in contrast, only occurs with the lowercased lemma 'sie' in the training data and the latter is thus applied to any token with surface form 'sie' that is tagged as pronoun when the model is applied for decoding.

Since one of the two lemmas is not an eligible option for our algorithm, the other lemma is chosen by our algorithm in all ambiguous cases. If we insisted on using the capitalized lemma 'Sie', we merely would need to pre-tag any non-sentence-initial occurrence of the one of the capitalized word forms that corresponds to it. That approach would yield a sufficient number of cases to reliably disambiguate both

[39] As the formal 'Sie' coincides in person and number with the plural 'sie' *'they'*, we frequently find cases with 'Sie' in sentence-initial position that cannot be distinguished on the sentence level, that is, a translator (human translator or translation algorithm) would need to look at a broader context or any existing translation to decide whether 'Sie' refers to the third person plural or the second person polite pronoun.

cases. Although the 'sie'/'Sie' case is the single most frequent lemma ambiguity with approximately 125 000 occurrences (out of which 25 000 originate from the word form 'Ihnen'), we proceed with a lowercase 'sie' as lemma – as suggested by our algorithm – since we are primarily interested in the disambiguation of lemma options for content words.

Without the 61 cases of ambiguous pronouns, 53 cases with 25 different lemma ambiguities remain (see Table 3.5). Some lemma pairs show an explicit semantic relation (e.g., 'Antwort' *'answer'* and 'Antworten' *'(the) answering'*), some leave a trace of their evolution from the same word (e.g., 'Stunde' *'hour'* and 'stunden' *'to defer'*, the root word of the derived noun 'Stunden' *'(the) deferral'*) and some appear to be semantically unrelated (e.g., 'Arm' *'arm'* and 'arm' *'poor'*, the root word of the derived noun 'Arme' *'pauper'*).

In two of the 53 cases evaluated, the algorithm selects the wrong lemma: 'Armen' is lemmatized as 'Arm' instead of 'Arme' and 'Reisen' as 'Reise' instead of 'Reisen'. A closer inspection of the lemma distribution matrix yields that the German lemma 'Arm' is aligned to the English lemma 'arm' in 52 % of its occurrences, but also in 33 % of the cases with 'poor'. Other aligned English lemmas with more than a single occurrence are 'wing' (7 %), 'branch' (3 %) and 'Arms' (3 %). For Spanish, we see a similar distribution: 'brazo' *'arm'* (60 %), 'pobre' *'poor'* (31 %), 'pobreza' *'poverty'* (3 %), 'rama' *'branch'* (3 %) and 'arms' (untranslated) (3 %).[40] The reason behind the semantically mismatching translations 'poor' and 'pobre' (and 'pauvre', 'povero', etc.) is that 'Arm' is also used in expressions such as "Arm und Reich" *'the rich and the poor'* (less frequently "Reich und Arm"), "Umverteilung von Reich nach Arm" *'redistribution (of wealth) from the rich to the poor'* or "Reich gegen Arm" *'the rich against the poor'*. This formulaic use, always in conjunction with 'Reich' *'the rich'*, did either not appear in the training data or 'Arm' meaning 'the poor' was deliberately lemmatized with the same lemma as 'Arm' meaning 'arm'.

In the case of 'Reisen' *'travel'/'traveling'*, we have to do with a noun derived from a verb ('reisen') that is used considerably less frequent than its counterpart 'Reise' *'voyage'* (25 vs. 645 occurrences). Since both are semantically close, the same words are used in other languages to translate both of them. That is why the lemma distribution matrix exhibits virtually the same translation options but with a significantly lower probability for 'Reisen', except for Slovak as 'cestovanie' is predominantly aligned to compounds with 'Reise' in German (e.g., 'Reisefreiheit', 'Reiseverbot', 'Reiseverkehr').

[40] It turns out that 'Arms' in "die Initiative Everything But Arms" *'the initiative Everything but Arms'* has been lemmatized as 'Arm' *'arm'* due to it being a valid word form of 'Arm'.

Table 3.5 – List of 25 lemma ambiguities from our evaluation and their frequency in the corpus. The translations are geared to corpus examples if the respective lemma does occur in the corpus. Non-occurring lemmas are underlined. In those cases, our algorithm selects the other lemma option by default.

Word form	First lemma	Second lemma	Frequency
Abkommen	Abkomme *'descendant'*	Abkommen *'agreement'*	11 705
Antworten	Antwort *'answer'*	Antworten *'answering'*	2306
Arbeiten	Arbeit *'work'*	Arbeiten *'working'*	2202
Armen	Arm *'arm'*	Arme *'pauper'*	761
durchführen	durchfahren *'to drive trough'*	durchführen *'to conduct'*	1925
Fällen	Fall *'case'*	Fällen *'felling'*	4789
gebraucht	brauchen *'to need'*	gebrauchen *'to employ'*	752
gehört	gehören *'to belong to'*	hören *'to listen'*	1950
getroffen	treffen *'to meet'*	triefen *'to ooze with'*	4883
gewährt	gewähren *'to concede'*	währen *'to last'*	1982
Gründen	Grund *'reason'*	Gründen *'founding'*	6544
Listen	List *'ruse'*	Liste *'list'*	422
Mitteln	Mittel *'means'*/*'average'*	Mitteln *'averaging'*	3632
Morden	Mord *'murder'*	Morden *'murdering'*	124
Rechte/Rechten	Recht *'right'*/*'law'*	Rechte *'Right'* (political)	14 933
Regeln	Regel *'rule'*	Regeln *'regulating'*	5437
Reisen	Reise *'voyage'*	Reisen *'travel'*/*'traveling'*	608
Stellen	Stelle *'position'*	Stellen *'positioning'*	2072
Streben	Strebe *'strut'*	Streben *'pursuit'*	577
Studien	Studie *'study'*	Studium *'academic studies'*	1354
Stunden	Stunde *'hour'*	Stunden *'deferring'*	1514
Summen	Summe *'sum'*	Summen *'humming'*	501
Tasten	Taste *'key'*	Tasten *'groping'*	8
Teilen	Teil *'part'*	Teilen *'dividing'*	2034
Zielen	Ziel *'goal'*	Zielen *'targeting'*	2322

3.2.2 Particle Verbs in German

In the previous section, we assumed that a lemma can be assigned to each token and that, conversely, the unit a lemma should be assigned to is a single token. This assumption is adequate for a predominant number of cases. There are, in principle, two possible deviations from it: On the one hand, two or more lexical units constitute a token as in German compound nouns 'Diplom-Studiengang' *'diploma degree course'* or '24-Stunden-Bereitschaft' *'24 hours standby'*, English

adverbial expressions such as 'turn-of-the-century' as in 'turn-of-the-century magic' or 'better-than-expected' as in 'better-than-expected tax revenues' and Portuguese idiomatic expression 'bicho-de-sete-cabeças' *'rocket science'*, which is a spelling variant of 'bicho de sete cabeças' without hyphens. On the other hand, a lexical unit can be represented by two or more tokens. This applies, for instance, to German and Dutch particle verbs.[41]

Particle verbs, or separable verbs, as they are often referred to, show a long record of investigation in respect of diverse aspects (see, for instance, Booij 1990; Lüdeling 2001; Zeller 2001; Müller 2003; Roßdeutscher 2011; Bott and Schulte im Walde 2015; Dewell 2015). Those verbs are characterized by a prefix particle that is either attached to its base verb or detached from it, depending on syntactic conditions.[42] In the latter case it is realized as single token in the same sentence, typically on a position following the verb.

The detached particle can exceptionally precede its corresponding verb when topicalized (see Zeller 2001; Volk, Clematide et al. 2016). These cases are considerably less frequent; in sentence-initial position, we only see the particles 'hinzu' (534 cases), 'fest' (100 cases) and 'los' (1 case) from the verbs 'hinzukommen' *'to supervene'*, 'feststehen' *'to be certain'* and 'losgehen' *'to start'*. These occurrences, unlike Zeller's examples, do not contrast two particle verbs that share the same base verb, but function rather as formulaic discourse elements (e.g., "Hinzu kommt der Menschenhandel mit burmesischen Mädchen nach Thailand zu Prostitutionszwecken." *'Then there is the traffic in young Burmese girls sold into prostitution in Thailand.'*; "Fest steht, dass noch viele Fragen unbeantwortet sind." *'It is a fact that many questions have still not yet been answered.'*; "Fest steht: Wenn diese Gesetze Anwendung finden, werden besonders europäische Unternehmen Schaden nehmen." *'What is certain is that if these laws are applied, European undertakings in particular will be damaged.'*).

In what follows, we focus on the vast majority of prefix particles following their base verbs and describe the method we use for reestablishing the link between the two. Knowing this link enables us to reconstruct the correct lemmas of those particle verbs. The lemma assigned to the base verb token is – following the schema of one lemma per token – the one of the base verb. On the one hand, this is owed to the tagging and lemmatization algorithm not being aware of the connection between base verb and separated prefix particle. On the other hand, it is questionable whether assigning the lemma of the actual verb to the base verb would be beneficial to all subsequent tasks since that lemma corresponds effectively to two tokens and not only one.

[41] For us lacking expertise in Dutch and the Dutch tagset lacking a dedicated tag for prefix particles of particle verbs, we limit ourselves to particle verbs in German.

[42] "This happens in matrix clauses when the verb is finite and occurs in present or past tense, or when the verb is in imperative form" (Volk, Clematide et al. 2016)

A reason for substituting the lemma of the base verb by the lemma of the particle verb is that particle verbs in German frequently differ in meaning from their base verbs (e.g., 'stellen' *'to put'/'to place'* vs. 'darstellen' *'to represent'/'to illustrate'*, 'geben' *'to give'* vs. 'preisgeben' *'to relinquish'/'to divulge'*, 'fallen' *'to fall'* vs. 'auseinanderfallen' *'to diverge'/'to fall apart'*, 'schlagen' *'to beat'/'to hit'* vs. 'vorschlagen' *'to propose'/'to suggest'*). Our solution is to maintain the relation between each two tokens of separated particle verbs together with the lemma of the recombined parts. This enables us to decide how to proceed depending on the respective task.[43]

Identifying Detached Particles and their Base Verbs

We employ the reattachment algorithm described in (Volk, Clematide et al. 2016) for identifying combinations of base verb and prefix particles. The approach consists of two steps:

First, for each separated verb prefix that has been identified in the part-of-speech tagging step, we look backwards in the sentence for a finite full verb or imperative verb.[44] If there is more than one match, we chose the rightmost one, that is, the one with the smallest distance to the particle. That way, we are certain not to choose the wrong verb from a set of coordinated verbs. However, we consequently miss cases where another verb (e.g., in a subordinate clause) is located between base verb and prefix particle.[45] The distance between base verb and prefix particle can be arbitrary long, oftentimes ranging over the whole sentence[46]

[43] Though not having implemented it, we expect that replacing the base verb with the actual verb's lemma and omitting the prefix particle would improve the performance of bilingual word aligners (see Section 4.4.1), in particular since this manipulation would reduce the alignment to be identified to a one-to-one correspondence between two verbs in many cases (e.g., 'darstellen'/'represent', 'abzeichnen'/'emerge' or 'durcheinanderbringen'/'confuse'). In a subsequent reconstruction step, all alignment units including the verb would need to be extended to also include the prefix particle (e.g., to convert the one-to-one into a one-to-two alignment).

[44] Imperatives are used in the European Parliament's debates rather infrequently. We only find 718 verbs in imperative mood in our corpus (e.g., "Gebt uns einen einzigen Arbeitsort, einen einzigen Sitz in Brüssel." *'give us one workplace, a single seat in Brussels.'*).

[45] "Herr Präsident, ich schließe mich den Bemerkungen der Berichterstatterin, soweit sie die sozialpolitische Komponente der Münzen anbelangt, voll inhaltlich an." *'Mr President, I fully endorse the substance of the rapporteur's comments in so far as they referred to the social aspect of the coins.'*

[46] "Wir verweisen diese Angelegenheit somit an den Haushaltsausschuss und die anderen, zur Abgabe einer Stellungnahme aufgeforderten Ausschüsse, d.h. den Ausschuss für auswärtige Angelegenheiten und den Ausschuss für Industrie, Außenhandel, Forschung und Energie zurück." *'We therefore refer the subject back to the Committee on Budgets and to the committees which are to issue opinions on this subject, that is to say the Committee on Foreign Affairs, Human Rights, Common Security and Defence Policy and the Committee on Industry, External Trade, Research and Energy.'*

– except for the prefield/initial field (German *'Vorfeld'*) position pursuant to the theory of topological fields (Herling 1821).

Having identified a candidate pair of base verb and prefix particle, we decide in a second step whether the two of them fit together as particle verb. To this end, we look up the hypothetical lemma consisting of the prefix attached to the base verb's lemma in the list of known lemmas from our corpus. If it is not to be found, we have no evidence for the existence of the hypothesized verb and therefore discard the candidate. That way, we prevent combinations of the prefix with a wrong verb, for instance, with the finite verb of a subordinate clause (see above). At the same time, we discard valid pairs that never occur in their infinite form. This is, for instance, the case with 'beistimmen' *'to agree'*: "Ich stimme den durch diese Entschließung eingeführten Änderungen bei" *'I agree with the amendments introduced by this resolution'*.

The method described so far corresponds to Volk, Clematide et al.'s approach, with the exception that we do not perform morphological analysis of the unknown hypothetical particle verbs. A random sample of 200 identified particle verbs yields four errors, which corresponds to a precision of 0.98. Two of them are due to wrong part-of-speech tags, one is entailed by a relative clause between the prefix particle and its base verb, which leads to assignment of the wrong verb, and, in the fourth case, the prefix particle occurs in a subordinate clause preceding its base verb, exceptionally without being attached to it: "Abschließend halte ich es in Anbetracht der Zielsetzung der Präferenzregelungen für wichtig, daß diese auch den am wenigsten entwickelten Ländern zugute kommen." *'Lastly, given the aim of the preferential arrangements, I think it is important that they should also benefit the least developed countries.'* Here, the part-of-speech tagger correctly identifies the prefix particle but the reattachment algorithm consequently searches for the base verb in the wrong direction and proposes 'zugutehalten' *'to make allowances for sb.'s sth.'*, which, though being rare, does exist in the non-separated form in our corpus.

In addition to these errors from our evaluation sample, we also find – infrequent – cases where an alleged prefix particle attached to the lemma of the preceding finite verb yields a prevalent particle verb, however with a different particle-verb boundary (e.g., prefix particle 'her' attached to 'ankommen' *'to arrive'* gives 'her-ankommen' *'to reach'*/*'to approach'*, which is actually composed of the prefix particle 'heran' and 'kommen' *'come'*).[47] This is also limited to sentences where the preceding verb is located in a subordinate clause.

[47] "(…) den Zeitpunkt zu vermerken, wann etwas abgegangen ist, sowie den Ort, an dem es ankommt, so daß man von der Zollkontrolle her genau eruieren kann, wann das Gut abgegangen ist und wo es eingegangen ist (…)" *'(…) to ensure that a note is made of the departure time of consignments and their destination, so that customs checks can reveal precisely when the goods left and where they were delivered (…)'*

Efforts to Correct Erroneously Identified Particle Verbs

Parting from the verb and particle pairs found in the first two steps, we aim at identifying those that do not actually belong together (i.e., the false positives) by evaluating the word alignments of the base verb. Our underlying assumption is that the base verb alone and the composite particle verbs typically differ in meaning and can thus be expected to hold a discriminative alignment distribution. We revert for this purpose to the lemma distribution matrix explained in Section 3.2.1, which we update by the particle verbs identified so far. That way, we reduce the proportion of base verb lemmas erroneously being counted as correspondences of particle verb translations.[48]

Due to constellations that the algorithm does not handle (see above), it is, however, in many cases not possible to identify all pairs of base verbs λ_b and prefix particles λ_p and, consequently, the lemma distribution matrix still holds some probability mass for translations of the base verb. Since base verbs are typically common verbs with high frequencies in our corpus while particle verbs show considerably lower frequencies,[49] this probability can still outclass the probability of proper translations.[50] The ratio of these two probabilities, $p_a(\lambda_b|\lambda_{b'})$ for the probability of a foreign lemma b' and $p_a(\lambda_{p+b}|\lambda_{b'})$, which is shown in Equation 3.4, is thus not a reliable indicator for distinguishing between the raw base verb and a particle verb built on it as long as we do not account for these erroneous cases in the lemma distribution matrix.

$$r_a(\lambda_b, \lambda_{p+b}|\lambda_{b'}) = \frac{p_a(\lambda_b|\lambda_{b'})}{p_a(\lambda_{p+b}|\lambda_{b'})} \tag{3.4}$$

A characteristic of German particle verbs is that, when their composition is semantically transparent, we often find phrasal counterparts in the other languages, in particular in closely related languages (i.e., Germanic, but also Romance languages). The German particle verb 'klarmachen' (*'clear'* + *'make'*) is, for instance, frequently translated to English as 'to make clear' and to Italian as 'mettere in chiaro', 'zurücklassen' likewise as 'to leave behind' and 'lasciare indietro'. In other

[48] This update reduces, for instance, the number of (erroneous) lemma correspondences of 'schlagen' *'to hit'*/*'to beat'* and English 'propose' from 2884 to 66, which, in turn, raises the alignment probability of English 'beat' given 'schlagen' from 0.016 to 0.341.

[49] An exception is, for instance, the verb 'hinweisen' *'to indicate'*/*'to reference'*, which is on average 40 times more frequent than 'weisen' *'to point'*/*'to show'* or *'beuten', which forms part of the particle verb 'ausbeuten' *'to exploit'* but does not exist anymore in German as an independent verb.

[50] While the probability of 'propose' given 'schlagen' has been lowered considerably, it is still 3.4 % higher than the probability of 'beat' given 'schlagen'. The predominance of the verb 'propose' over 'beat' can safely be attributed to the parliamentary origin of the corpus.

cases such as 'entgegenwirken', the corresponding verbs have the prefix incorporated (English: 'counteract'; Italian: 'contrastare'). In a parallel sentence, where the prefix particle is separated from the base verb in the German part, bilingual word aligners will generate two alignment units: one for both particles and one for both verbs. This is owed to the fact that the aligners' language models have learned the correspondence of, for instance, 'machen'/'make' and 'klar'/'clear' and the probabilities for 'machen'/'clear' and 'klar'/'make' will be significantly lower.[51] The preference of those word aligners for the most basic correspondence of one token in each language is depicted in Figure 4.21.

Another problematic case is when the base verb forms part of a multiword expression. The base verb of one of the false positives pairs, 'suchen' *'to search'*, belongs to the expression 'die Fehler bei anderen suchen', literally *'to search for faults at others'*, which translates to 'to blame others' in most other languages.[52] The lemma alignment distribution shows no evidence for the correspondence of 'to blame' on any language and either 'suchen' or the (in this case) wrongly recomposed particle verb 'absuchen' *'to scan'*/*'to search'*.

We try to derive a heuristic method from several statistical values associated with each case of identified pairs. To this end, we compile a small sample of true and false positives (10 each) of the identified pairs and perform linear regression analysis on it. The statistical values that we analyze originate from the lemma distribution matrix and the lemma alignment distribution overlap (see Section 5.1 and Appendix C.1) of base verb, particle verb and the respective aligned words (see Section 4.5.1). In addition, we include raw lemma frequencies for the verbs in question and the number of aligners supporting a particular alignment.

None of the above mentioned values alone and no linear combination of them can tell apart the true positives from the false positives in our small sample according to the analysis. Also, none of the resulting linear equations could at least explain half of the data points ($r^2 < 0.5$), which means that the chosen features are not helpful for telling apart the two possible cases. We assume that if we cannot find a good fitting model for our small sample, we will neither be able to find a heuristic method to reliably identify false positives based on those statistical values.

[51] Phrasal correspondence cannot be learned either as the parts of the phrasal/particle verbs in both languages do not frequently appear together.

[52] German: "Ja, es stimmt doch, Sie suchen gerne die Fehler bei anderen"; English: "Yes, it is true, Martin, you like blaming others"; Italian: "Sì, è vero, Martin, a te piace incolpare gli altri"; Spanish: "Sí, es verdad, Martin, te gusta culpar a los demás"; Slovak: "Áno, je to pravda, Martin, vy radi obviňujete iných"; Slovene: "Da, res je, Martin, da radi krivite druge"; Swedish: "Jo, det stämmer, Martin, du tycker om att skylla på andra"

Limitation

Apart from the limitations described in (Volk, Clematide et al. 2016), namely disregarding topicalized prefix particles, coordinated prefixes (not discussed here) and cases with interfering verbs of clauses in between the two elements of a particle verb, we are also not capable of detecting particle verbs with two separable prefixes,[53] for instance, 'wiederherstellen' *'to restore'/'to recover'* or 'wiederaufstehen'/'wiederauferstehen' *'to rise from the dead'*.[54] In these cases, the resulting lemma excludes the second prefix 'wieder' *'again'*.

3.3 Dependency Parsing

We employ the *MaltParser* (Nivre, Hall et al. 2006) to derive syntactical dependency relations in English, French, German, Italian, Spanish and Swedish. The MaltParser website[55] provides pre-trained language models for a couple of languages of which we use the English and Swedish ones.

Although recent experiments have shown that parsers without explicit part-of-speech tags achieve good results (Vinyals et al. 2015), most parsers rely on part-of-speech tagged input. In order to generate reasonable output, the tagset used by tagger and parser need to agree. This is not the case for the part-of-speech English models provided by the TreeTagger and MaltParser. Even though both build on the Penn Treebank tagset (Santorini 1990), different versions have been used for training so that we need to adapt the tags generated by the TreeTagger model to the ones that the MaltParser model expects.

Our MaltParser model for German has been trained on the TüBa-D/Z treebank (Telljohann, Hinrichs, Kübler et al. 2003; Telljohann, Hinrichs and Kübler 2004), which utilizes the STTS tagset, as do both part-of-speech taggers we apply to German. It generates dependency relations which are labeled according to the Hamburg Dependency Treebank (Foth et al. 2014; Foth 2006). The Swedish parsing model has been trained on the Swedish Treebank (Nivre and Megyesi 2007), which is the union of two treebanks: Talbanken (Einarsson 1976) and SUC (Ejerhed and Källgren 1997; Gustafson-Capková and Hartmann 2006). The de-

[53] In contrast to verbs with two prefixes where only one is separable (e.g., 'anvertrauen' *'to entrust'* as in "Warum vertrauen wir nicht die Verwaltung der eventuell von uns erzeugten Überschüsse einer Europäischen Agentur an […]?").

[54] "Wir knüpfen Bande und stellen sie wieder her, wenn sie zerreißen. *'We have woven ties which we mend when they become frayed.'*; "Manche arme Menschen stehen von den Toten wieder auf, sobald ihre Organe in lebende Menschen eingepflanzt sind" *'Some poor people positively rise from their graves, in a manner of speaking, when their organs are transplanted into living people.'*

[55] http://maltparser.org/

pendency labels used correspond to the MAMBA annotation scheme (Nilsson and Hall 2005; Teleman 1974). The English model generates Stanford typed dependencies (Marneffe and Manning 2008), which served as model for the universal dependency relations (Marneffe, Dozat et al. 2014).

We use the annotation pipeline by (Baffelli 2016) in FEP9, which has originally been developed for Italian, in an adapted version also for French and Spanish. Like in the original Italian pipeline, the French and Spanish parsing models generate relations labeled by version 1 universal dependency labels (Marneffe, Dozat et al. 2014; see also McDonald et al. 2013). Universal dependency relations comprise a fix set of relation labels for syntactic dependencies that are seen by the authors as universally applicable to any language. They are supplemented by language-specific dependency labels, which only apply to particular languages or groups of languages. In Appendix A.1, we list both universal and language-specific dependency labels and indicate if they appear in the relations generated by the respective parsing models for French, Italian and Spanish.

All three languages, French, Italian and Spanish, are morphologically analyzed with Morfette (Chrupała et al. 2008) and the results are fed to the respective MaltParser model, which is supposed to support the parser in taking the right decision by resolving ambiguities. In the Italian pipeline, clitics are separated from their corresponding verbs and treated as single tokens. After parsing, we rejoin them and drop any dependency relation of the clitics. The verb form 'occuparsene' *'to deal with it'*, for instance, is split into 'occupare *'to occupy'*, 'se' (non-standard form of reflexive pronoun) and 'ne' (pronoun coreferring to something previously mentioned). After parsing, we treat the separated verb as if it had been parsed as the compound form with both clitics attached.

For an earlier version of our corpus (FEP6), we built our own MaltParser model for Italian.[56] To this end, we obtained the Italian Stanford Dependency Treebank (ISDT)[57] from the evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA)[58] and replaced tags and lemmas with the respective fields returned by the Italian TreeTagger model. We added universal part-of-speech tags by mapping the tagset used by the TreeTagger model (see Section 3.2). We continued by training a parsing model on the modified treebank using the MaltOptimizer (M. Ballesteros and Nivre 2012). This method adds tagging errors to the gold data and thus leads to a performance loss compared with a model trained on the unmodified treebank, which can only be applied to input data with the same tagset. Furthermore, the lack of morphological information in

[56] The resulting dependency relations in FEP6 have been used in several works (i.a. Graën 2017; Graën and Bless 2017). Parsing in English and German has been performed with the same models as in FEP9; no other language has been parsed for FEP6.

[57] http://medialab.di.unipi.it/wiki/ISDT

[58] http://www.evalita.it/

the TreeTagger output is also likely to have decreased parsing accuracy for Italian in this corpus version.

In summary, we have syntactically parsed 6 of the 16 languages present in the latest version of our corpus. Altogether, the parsing models use four different label sets to distinguish the kind of syntactic relations between token pairs. With regard to performance on our corpus data, we can at most expect the numbers reported on the test sets of the respective training corpora (e.g., a label attachment score of 86 % reported by (Baffelli 2016)), if those numbers are available at all. If they are not, we can safely assume them not to be significantly higher than other parsers' performance, which is, after all, still considerably lower for syntactic parsing than for part-of-speech tagging, first of all due to the complexity of the problem.

Although it would be helpful for comparison of syntactic structures in parallel sentences, if all language models were using the same set of dependency relations (i.e., the universal ones), we only use them to identify elementary relations such as verb complex and noun phrase parts (Section 4.5), direct objects (Section 5.3) and prepositions (Section 5.4). These relations are typically less ambiguous (see, e.g., ibid., Section 5.2.3.4).[59] Wherever our work relies on syntactic relations, we either require parallel constellations, that is, that partial parsing structure agrees between two languages, use it for statistical analyses or use it as supporting evidence. In all cases, we reduce the error probability of our respective application by not letting it rely on single instances of dependency relations.

3.4 Database Corpus

The corpus as described in the previous sections makes use of a limited number of relational data types, that is, relations between entities and attributes. The central element of our corpus schema is the token entity. Unlike Chiarcos, Ritz et al. (2012), we do not allow for alternative layers of tokenization. For applying the pre-trained models we make use of (i.e., the ones used by TreeTagger, TnT, Stagger, Maltparser and Morfette) it is important to provide input data that the respective model knows to handle. For tokenization, that means that we should not tokenize "doesn't" as [doesn]['][t] if the model expects [does][n't] or, otherwise, we have to expect degraded performance. This is why we transform the input (e.g., conversion of the typographic apostrophe to the typewriter one) and output (e.g., reattachment of Italian clitics) of the tools we employ where necessary.

[59] The conversion of language-specific dependency relations to universal ones is not as simple as mapping language-specific part-of-speech tags to universal part-of-speech tags (see Section 3.2); such a conversion would involve transformations in the dependency tree.

In (Graën and Clematide 2015, Section 3), we sketch the data structures needed to represent processed text corpora. In the latest version of our corpus, we only use three basic types alongside the core structure consisting of tokens, sentence segments and texts:

1. *Attributes*: Part-of-speech tagging and lemmatization add an attributive layer to the sequence of tokens. Multiple layers of the same kind are possible (e.g., different part-of-speech taggers applied to the same token sequences, as we did for German).

2. *Directed binary relations*: Syntactic dependency parsing adds a layer that relates token pairs with attributes (i.e., the dependency labels). Multiple layers are possible, too, though we do not use different parser or parsing models for the same language. This type can also be used to model other intralingual relations such as coreferences.

3. *Sets*: Multilingual alignments, that is, the correspondence of elements in multiple languages, on both sentence and token level require a structure that relates an arbitrary number of elements. Hierarchy of alignments can be expressed by means of inclusion. In contrast to dependency relations, multilingual alignments are by definition symmetric.

The interval configuration we list in (ibid.) is needed for any annotation that identifies continuous structures such as syntactic constituents. In our corpus annotation, we rely on dependency parsing and do not need support for continuous components. The components identified by our multilingual alignment word approach (Section 4.5) are frequently discontinuous (e.g., German "Dies zieht die Vereinbarung in Zweifel" *'This calls into question the agreement'*) and thus cannot be represented as (aligned) intervals.

Advantages of Relational Database Management Systems

Relational database management systems (RDBMS) are built on relations and set operations. They dispose of sophisticated indexing techniques (e.g., functional indices, concatenated indices and indices on letter trigrams for partial string matching) that allow for efficient retrieval of the data stored in a database (for an overview see Winand 2012). The database schema, that is, the entities represented in the database and their relations with each other in its entirety, is defined by the user. It is consequently the user's duty to derive a model that comprises all relevant data and relations for the envisaged application or applications

in a non-redundant way.[60] Once set up and populated with data, the database can be queried for any data using arbitrary combinations of entities permitted by its schema. To this end, the required data is functionally described in relational algebra, which is typically done by variants of the SQL standard.[61]

Our decision to use a relational database for storing and querying our corpus is primarily motivated by these insights. The idea is not new, though: (Davies 2005) explored the suitability of an RDBMS for large corpora with several layers of annotation more than a decade ago and found speed of retrieval, extensibility and flexibility of corpus retrieval by means of SQL queries the most compelling reasons in favor of corpus databases. Our RDBMS of choice, PostgreSQL (PostgreSQL Global Development Group 2017),[62] is able to efficiently handle data loads that are several orders of magnitude larger than what the biggest text corpora available with hundreds of annotation layers would require and to process far more complex queries than those we need for corpus retrieval. Other specialized corpus query systems, the most prominent of which presumably is the *Corpus Workbench (CWB)* (Christ 1994; Evert and Hardie 2011) with its *Corpus Query Processor (CQP)*, typically rely on self-made indexing solutions using plain files.[63]

SQL is a powerful query language, capable of retrieving and statistically analyzing any data that is even remotely related. Since SQL queries express a functional description of the data to be retrieved, the database's query planner (see Winand 2012, Chapter 2) evaluates the estimated costs (i.e., the expected times that each action will require) of different query execution plans that are equivalent with respect to the result. It subsequently chooses the fastest one.

This implies generating a list of possible equivalent query execution plans for the query in question, estimating their costs by taking into account numerous parameters such as selectivity of query parts (by means of distribution statistics for accessed attributes), available indices and the latency of the underlying storage devices.[64] Non-selective aggregating queries can only be optimized up to the extent of the effective costs to traverse the whole dataset that is to be analyzed. In those cases (e.g., for calculating collocation scores), it is advantageous to precalculate the required values and index them as well. PostgreSQL provides an extension to the SQL standard that implements materialized views, that is, a physical repre-

[60] While formal rules for database normalization exist and a particular degree of normalization is typically required to prevent anomalies (i.e., inconsistencies) in the database, the deliberate introduction of redundancies may be licensed by performance requirements.

[61] ISO/IEC JTC 1 2016.

[62] http://postgresql.org/

[63] ANNIS (Chiarcos, Dipper et al. 2008; Krause and Zeldes 2014) also partly relies on PostgreSQL for data storage and querying.

[64] Smith (2010) deals with performance tuning of the PostgreSQL DBMS to make queries more efficient and allow for the query planner to estimate costs more accurately.

sentation (unlike views) of the query results for a particular point in time.[65] Since we do not add primary data to our database corpus once populated, we do not run the risk of accessing outdated values.[66]

As versatile and expressive SQL queries are, they are arguably not the query language of choice for general linguistic questions (Graën and Clematide 2015). On the one hand, they tend to be verbose and repetitive as the schema-inherent relations have to be expressed explicitly each time. On the other hand, these relations have to be known by whoever composes a query, which renders composing them overly difficult for most users. The obvious solution to these problems is to have the SQL be generated as interim query language starting from a query language that requires the user to only describe linguistically motivated entities and their relations, a similar approach to the one ANNIS (Chiarcos, Dipper et al. 2008; Krause and Zeldes 2014) has taken (Rosenfeld 2010; see also Clematide 2015).

In contrast to corpus query tools like ANNIS, our work does not aim at supporting arbitrary corpus queries by users. For our applications (see Chapter 5), which combine word alignment with one or more annotation layers, we use corpus query templates that are combined to compose the final query at runtime (see, for instance, Graën, Sandoz et al. 2017). The conversion of queries expressed in an advanced query language that, in addition to multiple layers of annotation, also allows for referencing multilingual alignments (Section 4.5) into SQL queries is a major challenge, which cannot be addressed here.

[65] https://www.postgresql.org/docs/10/static/sql-creatematerializedview.html

[66] In case we wanted to add corpus data to an existing database corpus, we would simply need to refresh the materialized view (i.e., to repopulate the relation with up-to-date data).

Chapter 4

Alignment Methods for Parallel Text Corpora

This chapter deals with different levels of alignment, that is, the identification of corresponding elements on various structural levels of parallel corpora. For a comprehensive overview, we refer the reader to (Tiedemann 2011, Chapter 2). The chapter descends thematically from the alignment of bigger textual units to the comprehensive field of word alignment, always taking into account the particular challenges of consistent multilingual alignment. **Multilingual alignment** is the identification of corresponding elements in more than two languages simultaneously. Simard (1999) comments on alignment as **translation equivalence**: "Translation equivalences can be viewed at different levels of resolution, from the level of documents to those of structural divisions (chapters, sections, etc.), paragraphs, sentences, words, morphemes and eventually, characters."

Text alignment (Section 4.1) is a – potentially hierarchical – alignment of a parallel corpus that yields corresponding textual units, primarily based on meta-

Contributions

Mathias Müller and Ventsislav Zhechev improved our pipelines for bilingual word alignment. Christof Bless built the graphical alignment tool that we used for preparing our multilingual sentence and word alignment gold standards. The word alignment gold standard in six languages is the work of Selena Calleri and Barbara Pejkovic.

Both approaches to multilingual sentence and word alignment have been designed and implemented by the author. The same applies to their evaluations.

data and document structure. Whether a hierarchy is given or not depends on the structure of the respective corpus material. For further processing, only the minimal corresponding textual units identified at this level are relevant as they define the boundaries for subsequent sentence alignment. Multiparallel corpora (see Section 2.3) often already specify correspondence on the level of documents (e.g., resolutions in the UN corpus (Rafalovitch, Dale et al. 2009)).

Sentence alignment identifies corresponding sets of sentences, which are typically sequential ranges, in the respective languages. The additional complexity that **multilingual sentence alignment** (Section 4.3) introduces to the alignment process stems from the requirement of coherence, that is, the need for alignment boundaries in all particular languages to agree. Alignment boundaries that have no counterpart in at least one of those languages require **hierarchical sentence alignment**. From that hierarchy, we can extract **minimal alignments** for any subset of languages comprised.

The kind of alignment arguably most dealt with in literature is **word alignment** (Section 4.4).[1] Word alignment has not only driven machine learning approaches for statistical machine translation but also a series of more linguistically motivated tasks, such as the extraction of translation equivalents for multiword expressions. **Sub-sentential alignment** emanates from the aligned words aiming at the alignment of higher level, often syntax-related units.

As in multilingual sentence alignment, **multilingual word alignment** (Section 4.5) requires the alignments to be represented as a hierarchy since the parts of a complex unit, for example, a multiword expression, may consist of smaller corresponding units in only a subset of the languages being aligned. Multilingual word alignment thus acts as a junction of bilingual word and sub-sentential alignment for the alignment of words in three or more languages.

We expect hierarchical multilingual word alignments to be useful for comparison of linguistic phenomena between several languages. Language learners who have already acquired knowledge in another language benefit from multilingual corpus examples. The use of aligned single words and complex expressions in one or more **additional languages** can reveal structural and lexical similarities and differences.

Terminology

The term *alignment* has been used in literature to denominate four different concepts: First, it can refer to a particular alignment method (e.g., word alignment). Second, it can refer to the process of aligning parallel data (e.g., "the alignment

[1] Word alignment identifies corresponding tokens in a sentence and, from a today's perspective, should thus have better been called token alignment instead. In this chapter, we speak of words instead of tokens to adhere to the well-established terminology.

was carried out by …"). Third, the result of applying an alignment method to a particular unit (e.g., a sentence), a set of corresponding subunits (e.g., words), is typically called an alignment. Fourth, a single correspondence of those subunits (e.g., two words in two different languages) is also referred to as an alignment (e.g., a one-to-one alignment). These different meanings are often used side by side in publications (see, for instance, Varga et al. 2005; Braune and Fraser 2010; Abdul-Rauf et al. 2012). To avoid misreadings, we distinguish them by using the terms *alignment method*, *alignment process*, *alignment set* (AS) and *alignment unit* (AU),[2] respectively, whenever they are not determined otherwise. A manually created set of correct alignment units is called *gold alignment* and an application performing alignment is referred to as *alignment tool*, *aligner* for short.

The position between two adjacent alignment units is referred to as *alignment boundary*. Varga et al. (2005) base their evaluation on the list of consecutive alignment boundaries, which is admissible if the alignment set is *monotonic*, that is, its alignment units establish an order that corresponds to the order of sentences in both texts, and *complete*, that is, every original sentence forms part of one AU.

4.1 Text Alignment

Text alignment is the task of identifying minimal corresponding textual units in two or more languages in a collection of translations.[3] Unlike sentence and word alignment, text alignment depends for the most part on extra-linguistic properties, such as domain, origin and technical formatting of the textual material. If all language versions are close translations of each other, even paragraphs may be considered as smallest text AUs.

For book translations given as raw text, the obvious structure typically comprises (numbered) chapters and parallel texts will become correspondingly large. The correspondence of text may, however, not be given from the outset. For Tiedemann (2011), "[t]he first alignment task when building parallel corpora is to link corresponding document [sic] with each other." He refers to this task as document alignment, while Östling (2015) uses the term document linking. We typically expect a one-to-one correspondence of documents. Null alignments on the document level, that is, a missing translation for a particular textual unit in one language, are also possible. In fact, a considerable number of translated texts in the Europarl corpus (Koehn 2005) are missing (see Graën, Batinic et al. 2014).

[2] Brown, Lai et al. (1991) coined the term *bead* for an alignment unit in sentence alignment, but it did not prevail.

[3] The original text is not required to be part of the collection, though; all texts may well be translations of another source not comprised by the corpus. We will equally refer to them as translations.

In the two multilingual corpora compiled at our institute, Text+Berg (Göhring and Volk 2011) and the Credit Suisse Bulletin corpus (Volk, Amrhein et al. 2016), the units to be aligned at the document level are articles (see also Section 2.3). Null alignments pose a particular problem since missing translations of articles are not indicated in the respective languages (ibid., Section 4).

Ribeiro et al. (2000) propose a method to further subdivide parallel texts by detecting corresponding homographs such as proper names or numbers. They assume that those corresponding tokens are only valid when found at approximately the same relative position in the respective texts and filter out unreliable correspondences using a so-called 'confidence band' around the calculated linear regression line. Unlike a static confidence interval, the confidence band's width also depends on the relative position in the texts. Kay and Röscheisen (1993) are convinced that "long texts can almost always be expected to contain natural anchors, such as chapter and section headings, at which to make an a priori segmentation."

In the case of the Europarl corpus (Koehn 2005), three structural levels inherent to the plenary debates of the European Parliament are represented in the corpus compilation:

1. the partition into plenary sittings,[4]
2. the division of each sitting into thematic chapters, and
3. the subdivision of each chapter into speaker contributions or turns.

While the first level is indicated by the respective filenames, the latter two are numbered consecutively for each sitting. Unfortunately, the numbering of turns is broken oftentimes, which will lead to errors if we base our text alignment on them. In the best case, we could find discrepancies using some measurement on the aligned turns and reject those AUs and all subsequent ones of the same sitting to get rid of subsequent errors. In the worst case, we would not detect the erroneous AUs and continue applying sentence and subsequently word alignment on them, which would all be wrong.

These considerations – alongside with a number of other issues we discovered (see Graën, Batinic et al. 2014) – led to the creation of the Corrected & Structured Europarl Corpus (CoStEP), detailed in Section 2.3.1, on which we performed text alignment for speaker turns by exploiting meta information such as the speaker's name or political party. In a subsequent step, we removed those turns from the corpus that were available in one language only, which means that either we were

[4]Owed to the gradual enlargement of the European Union over time, there are generally fewer languages available for earlier dates. In addition, some translations are unavailable although the respective languages were official working languages of the European Union at the particular plenary dates.

unable to align them to corresponding turns in other languages or that they had no translations in the first place. Altogether, 162 400 speaker turns are available in CoStEP 1.0.

Furthermore, we obtained a list of parliament members from the European Union from which we added speaker attributes to the turns whenever we could identify the respective speaker in that list.[5] For identification, we relied on a fuzzy match of the speaker's names and a comparison of the respective sitting's dates with potentially matching members' mandates. In so doing, we were able to add metadata to 117 511 turns in CoStEP 1.0, out of which 102 622 made it into FEP9 (see below).

Our incentive for adding the speaker attributes was being able to distinguish native from non-native speakers. While we attain no absolute certainty when basing our assumption on the speakers' countries – especially not for multilingual ones –, we will presumably get a higher precision by, for example, excluding countries other than the United Kingdom and Ireland when looking for native English speakers.

For our final corpus, FEP9, we extracted all those turns from the latest CoStEP version (1.0) that are available in all our primary languages (English, French, German, Italian and Spanish; see also Section 1.1). We included translations in our secondary languages, whenever they were available. The resulting corpus comprises 146 544 parallel texts in all primary languages. The remaining languages form two groups with regard to the quantity of parallel texts as shown in Figure 4.1: one that covers more than three quarters of the primary languages' texts (Dutch, Finnish, Greek, Portuguese and Swedish) and one that covers less than one third (Bulgarian, Estonian, Polish, Romanian, Slovak and Slovene).

Earlier versions of our corpus were extracted from earlier versions of CoStEP and contained fewer texts in fewer languages: For FEP3, we extracted only the parallel turns of the primary languages from CoStEP 0.9.0. The successor, FEP6, is based on CoStEP 0.9.4 and comprises less parallel texts since, for that version, we required the primary languages' texts to be available in Finnish as well. In addition to Finnish, Polish translations were included whenever available.

4.2 Sentence Alignment

With aligned texts as a basis, the next step is to align corresponding sentences. If a translation closely resembles the original text, chances are that sentence boundaries coincide and thus the first sentence of one language corresponds to the first

[5] We added the member's forename, surname, the country she was representing and her political group in the parliament. In Europarl, and consequently in CoStEP, only one attribute 'name' was defined, which would refer to either a speaker's full name or her surname.

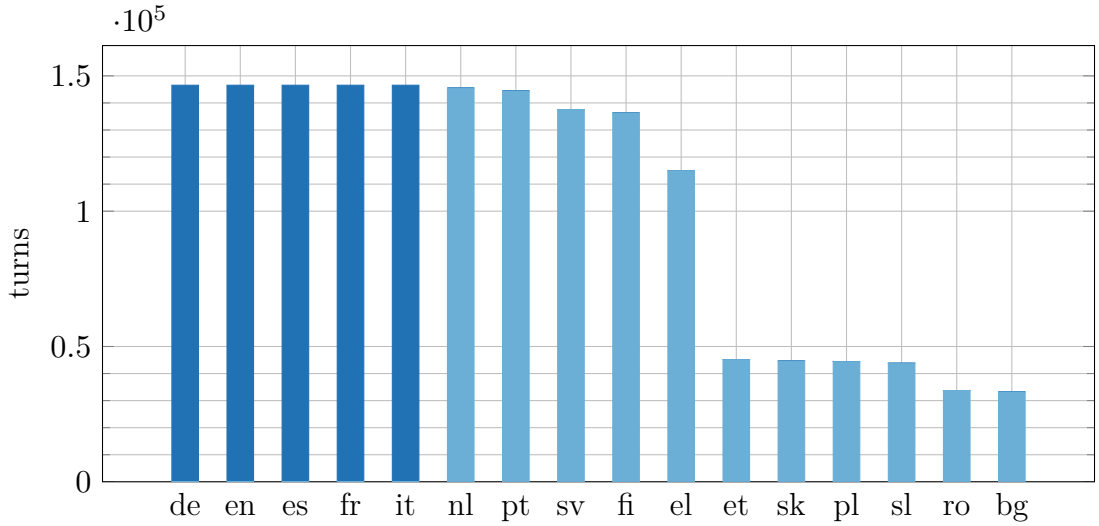


Figure 4.1 – Number of turns retrieved from CoStEP in the respective languages in FEP9. All turns are, by our design, available in English, French, German, Italian and Spanish. For space reasons, we use ISO 639-1 language codes.

sentence of the other language, and so forth. In this case of a series of one-to-one correspondences, depicted in Table 4.1, sentence alignment becomes a straightforward task.

Low error rates on sentence alignment already reported by early approaches (Brown, Lai et al. 1991; Gale and Church 1991) are best explained by the assumption that original sentence boundaries have been predominantly maintained by translators. The less frequent cases of one-to-many and many-to-many alignments are thus the ones that account for most of the errors observed. Gale and Church (1991, p. 85) report an error rate four times higher for two-to-one (2:1) than for one-to-one (1:1) alignments. They presume that the "low error rate is due to the high frequency of 1-1 alignments."[6] In their manually aligned test set of economic reports in English, French and German, one-to-one alignments account for 89 % of the total AUs, which is exactly the same proportion Ma (2006) found for Chinese-English AUs in their test set of United Nations' documents.

Apart from one-to-two alignments, that is, one sentence in one language corresponds to two sentences in another, a variety of other correspondence configurations exists. Their proportion among all AUs of a particular text depends first of all on how literally the text has been translated in terms of segmentation, that is, how much the translation's segmentation sticks to the original one. Hansen-

[6]Kay and Röscheisen (1993) attribute "near literalness" to the Canadian Hansard, an English/French parallel corpus containing the Canadian parliamentary debates, that has been used for evaluation by both approaches.

Table 4.1 – Sequence of five sentences in English, German and Spanish with concordant sentence boundaries. All texts are translations from French.

	English	German	Spanish
1	Of course, I have said it often before, I am no lover of capitalism.	Selbstredend bin ich, wie schon häufig gesagt, kein Freund des Kapitalismus.	Aunque por supuesto, como ya he dicho en otras muchas ocasiones, no soy un seguidor del capitalismo.
2	Capitalism is not an object of my affection, it is simply a means to an end.	Der Kapitalismus hat nicht meine Sympathie, er ist lediglich Mittel zum Zweck.	No es una de mis predilecciones, es simplemente un medio para conseguir un fin.
3	In any case, I do often like to distinguish between capitalism and liberalism.	Auf jeden Fall pflege ich oft zwischen Kapitalismus und Liberalismus zu unterscheiden.	En cualquier caso, a menudo me gusta hacer una diferencia entre el capitalismo y el liberalismo.
4	Clearly, my socialist friends are keen to combine these, yet the two things are not the same.	Meine sozialistischen Freunde werfen natürlich gerne beide zusammen, sie sind aber nicht das Gleiche.	Está claro que mis amigos socialistas tienden a combinarlos, pero se trata de dos cosas distintas.
5	Even I have to say it.	Das möchte ich doch einmal klarstellen.	Aunque tenga que decirlo.

Schirra (2003) summarizes Baker's (1996) translation hypothesis of simplification saying that "translators often break up long and complex sentences into two or more sentences in their translations in an effort to make the texts easier to read." If this hypothesis holds for translations between any pair of languages, we can expect a certain number of cases where two or more translations agree in segmentation having been simplified the same way.

Table 4.2 shows a correspondence configuration contrary to the one in Table 4.1. We can spot a one-to-three (1:3) alignment between English and German: Three individual English sentences correspond to two main clauses in German, joined by a dash, and a subordinate clause. The language pair Spanish-English shows a one-to-two (1:2) alignment: Two sentences in Spanish are connected by a conjunction ('y'). Finally, when we look at German and Spanish, we see a complex correspondence between two German and three Spanish sentences, a two-to-three (2:3) alignment.

In general, we may find untranslated sentences in parallel corpora. These so-called null alignments occur, among other reasons, when the translator decides to

Table 4.2 – Sentences in English, German and Spanish that require one-to-many (English/German and English/Spanish) and many-to-many alignment (German/Spanish). All texts are direct translations from French.

	English	German	Spanish
1	I hear MEPs who, I think, still believe in the effectiveness, honour and values of Europe, as well as feeling a certain pride in being European.	Europaabgeordnete, die meiner Meinung nach doch Grundsätze wie Effizienz und Ehre sowie die Wertvorstellungen Europas hochhalten und einen gewissen Stolz empfinden, Europäer zu sein – diese Abgeordneten höre ich ständig lamentieren und ein Sündenbekenntnis ablegen, dass an alledem im Grunde Europa schuld sei.	He escuchado las intervenciones de diputados al PE que, desde mi punto de vista, aún creen en la eficacia, el honor y los valores de Europa y que además sienten cierto orgullo de ser europeos.
2	I hear them constantly complaining and apologising.		Les he oído quejarse y pedir disculpas de un modo constante.
3	Basically this is all meant to be Europe's fault.		Todo esto significa esencialmente que es culpa de Europa y no puedo aceptarlo.
4	I do not accept that.	Dem stimme ich nicht zu.	

skip or add a sentence during translation, under the assumption that the target language audience does not require a given information or needs additional information to properly understand the text. Both phenomena may apply to, for example, newspaper articles; they do typically not appear in the debates of the European Parliament. However, translators frequently add additional information as shown in Table 4.3.

4.2.1 Approaches

Both approaches mentioned above (Brown, Lai et al. 1991; Gale and Church 1991) take advantage of the observation that the lengths of original and translated sentences correlate, that is, shorter sentences are translated with shorter sentences and longer ones with longer ones. Provided that a translation's purpose is to transmit the same information as the original text in a different language, this observation is underpinned by information theory (Shannon 1948). A mentionable exception are expressions that the translator chooses not to translate, but to accompany by a literal translation or description. This is typically the case for acronyms as depicted in Table 4.3. In those cases, the translation comprises more

information than the original. In the opposite case that some (minor) information is not transferred from source to target language,[7] the translation becomes relatively shorter than the original.[8] Both deviations have a negative impact on solely length-based sentence alignment approaches.

Table 4.3 – A description has been added to the English acronym 'PNR' for the German and Spanish translations. The original language is English.

	English	German	Spanish
1	We are currently working on a PNR package.	Wir arbeiten derzeit an einem Fluggastdatensatz-paket (Passenger Name Record, PNR).	En estos momentos, estamos trabajando sobre el paquete de registro de nombres de los pasajeros (PNR).

Sentence alignment algorithms for parallel texts assume strict monotonicity,[9] which means that the information conveyed follows the same order in the original and the translated text. This restriction, together with the principle of correlated lengths, suffices to attain a good overall alignment of parallel texts with automatic methods (more than 95 % correct AUs).

While the aforementioned alignment methods entirely rely on length information,[10] subsequent approaches also include lexical information. S. F. Chen's (1993) alignment model is based on a translation model that is learned from parallel data in a bootstrap approach. It starts by training a first model on 100 manually aligned sentence pairs and subsequently using it to align 20 000 new sentence pairs from the parallel corpus that is to be sentence-aligned. These alignments are, in turn, the basis for the training of a second model, which is used to align the whole corpus (including the previously aligned 20 000 pairs). He reports an improvement over previous, length-based methods.

Simard, Foster et al. (1993) introduce cognates (i.e., similar word forms in both languages) as source for alignment decisions and combine them with Gale and Church's (1991) length-based approach. Cognates work best for related languages that share parts of their vocabulary (see also Tiedemann 2011, Section 4.2). In the Europarl corpus, proper names and technical terms (e.g., 'CO2') are frequent cognates.

[7] Krynicki (2012) gives as motivation "[the] translator's decision not to render source-text material judged to be redundant or untranslatable."

[8] Relative length differences between languages are shown in Figure 4.6 in Section 4.3.1.

[9] For alignment algorithms on comparable corpora which need not comply with this restriction see (Plamada and Volk 2013), for instance.

[10] Brown, Lai et al. (1991) count tokens, whereas Gale and Church (1991) count characters.

Kay and Röscheisen (1993) contrast their approach with previously published length-based methods. As regards length, they only resort to the position of each sentence in both languages relative to the lengths of the respective texts. If the relative positions of a pair of sentences are close enough (the limit is the square root of the number of sentences available in the other language's text), this pair is considered "alignable" and consequently searched for words (they actually mean types, that is, surface forms of words) with a similar distribution. Those words are considered aligned if "the distributions of the words in their texts are sufficiently similar and if the total number of occurrences indicates that this pair is unlikely to be the result of a spurious match." Built on the obtained list of "word alignments", the sentence AS is inferred. The best AUs from that AS are then used as anchors and the process is repeated. In so doing, the coverage of the text that is aligned increases iteratively.

Later works (Moore 2002; Varga et al. 2005) employ a two-step approach: In a first step, sentence alignment is performed using the established length-based method. The so obtained AUs are then used to extract statistical lexical information which, in turn, serves as a basis for the second alignment process. Moore's solution is to first apply a length-based alignment method similar to (Brown, Lai et al. 1991) and to generate a simplified type 1 IBM word alignment model (see Koehn 2010, pp. 86–97) from the best resulting AUs. He then performs alignment with the generated model limited for performance reasons to those AU candidates that got a "nonnegligible" alignment score in the first run.

The aligner *hunalign* published by (Varga et al. 2005) uses a similar strategy. In an optional preparatory step, a dictionary is learned using a bootstrap approach: First, hunalign performs an initial sentence alignment based on identical word forms in both languages. Second, it derives the dictionary from frequently cooccurring word forms in high-scoring one-to-one sentence AUs. The dictionary bootstrapping is skipped if an external dictionary is provided by the user.[11]

In the actual alignment step, this dictionary and length information are employed together to align the parallel texts provided. The dictionary is used to "translate" the source language text into the target language.[12] Here, translation refers to the transformation of each token's word form into a word form of the target language by means of a dictionary lookup. If the word form in question is found in the dictionary multiple times, the target language word form with most occurrences in the target text is chosen. Word forms that are not found remain untranslated.

The number of identical word forms in a translated source sentence and a target sentence divided by the number of tokens of the longer sentence is hunalign's

[11] It is also possible to save the derived dictionary for later utilizing it when running hunalign.
[12] Source and target language are determined by the user.

measure for lexical similarity.[13] This similarity score and a character-based length ratio are calculated for all sentence pairs in a wide beam around the diagonal of the source and target language matrix[14] determined by relative positions. Dynamic programming is then used to determine the "optimal alignment trail" of one-to-one and one-to-two (including two-to-one) AUs. Remaining zero-to-one alignments are subsequently merged with existing AUs of adjacent sentences if that merge improves the length-based ratio in comparison to the original AU.

Braune and Fraser's (2010) aligner *Gargantua* also implements two alignment steps. Similar to (Moore 2002), it learns an IBM alignment model 1 in a first length-based alignment step. In contrast to Moore's and all other published approaches so far, their second alignment step first performs a limited alignment that only allows for each sentence in both languages to be aligned to one or zero sentences in the other language, followed by a clustering of the one-to-one and null alignments generated in this manner. They motivate their approach with targeting "asymmetric parallel corpora", that is, parallel corpora that show a large quantity of null alignments. While Gale and Church (1991) allow a maximum of two sentences per AU for each language, Braune and Fraser's approach is limited to identifying AUs with a single sentence in one language, that is, it can only identify one-to-one or one-to-many alignments.

bleualign (Sennrich and Volk 2010) is a sentence alignment approach that requires one of the two texts to be translated into the other text's language. Unlike hunalign's lexical lookup and replacement approach, bleualign makes use of a trained machine translation system. Once the translation has been generated, alignment is performed on two texts (the translation plus the other, untranslated text) of the same language using the BLEU metric (Papineni et al. 2002), which has been designed to measure the quality of machine translation systems. For a pair of unrelated sentences, the BLEU score is typically 0, which limits the number of potential alignments considerably compared to the number of possible combinations of source and target language sentences. The bleualign approach is thus particularly useful for texts with null alignments.

Multilingual Sentence Alignment

The aforementioned methods deal with the problem of aligning parallel texts in two languages. The availability of parallel corpora in more than two languages led Simard (1999) to ask: "Do they make new applications possible? Can methods

[13] Many matching numbers additionally increase the lexical similarity score.

[14] Since "at least a 500-sentence neighborhood is calculated or all sentences closer than 10% of the longer text", this corresponds to an exhaustive comparison for texts with less than 500 sentences in one of the languages.

developed for handling bilingual texts be applied to multilingual texts? More generally: is there anything to gain in viewing multilingual documents as more than just multiple pairs of translations?"

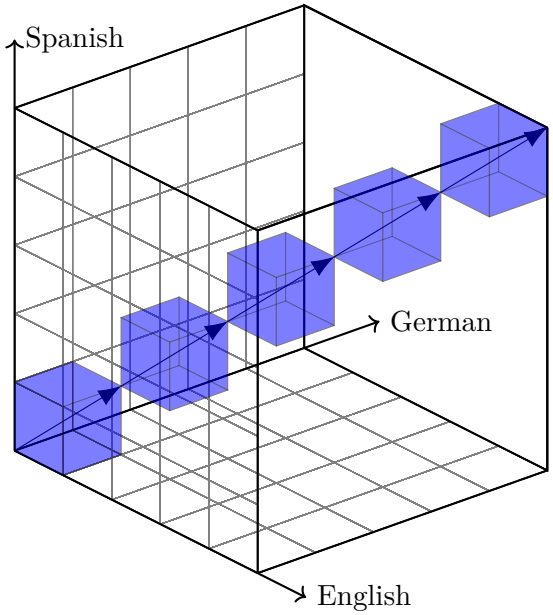


Figure 4.2 – Alignment units (AUs) depicted as hyperrectangles (here: cubes) in a discrete vector space (here: cuboids in a three-dimensional space). Each AU is described by a vector and the sequence of all AUs spans the entire space.

He formalizes multilingual alignment of n languages as the search for segmentation points in an n-dimensional vector space, where the discrete values of the axes correspond to sentence boundaries (i.e., potential alignment boundaries) in the respective languages. The alignment algorithm's task is "finding an optimal path in a rectangular matrix" (Simard 1999) and the sum of all segments represented by the vectors spanning them thus equals the correspondence of all n parallel texts. For the simple case of strictly parallel sentences in Table 4.1 (depicted in Figure 4.2), we formalize the one-to-one-to-one alignments as list of vectors $A = [(1,1,1)^T, (1,1,1)^T, (1,1,1)^T, (1,1,1)^T, (1,1,1)^T]$, which add up to the overall parallel texts with five sentences per language:

$$\sum_{\vec{a} \in A} \vec{a} = \begin{pmatrix} 5 \\ 5 \\ 5 \end{pmatrix}$$

The example in Table 4.2 can only be represented as one single alignment unit, namely $(4,3,2)^T$, due to missing coinciding intermediate alignment boundaries (depicted in Figure 4.3).

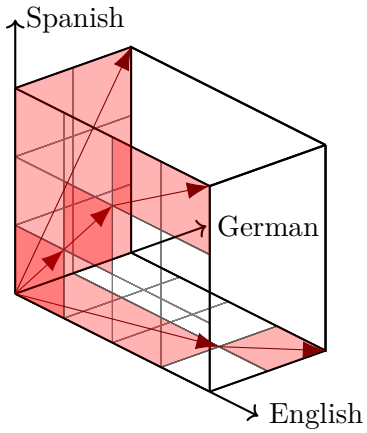


Figure 4.3 – Two-dimensional AUs for all three language pairs. The only possible AU for all three languages comprises the entire space and thus aligns all sequences of sentences.

Simard's (1999) approach to multilingual sentence alignment is to use bilingual alignments to construct the resulting multilingual ones. To this end, he rates all language pairs by the global alignment probability that a bilingual alignment method yields for the respective pair. As he exemplifies his approach with three languages (named A, B and C), there are three pairs of languages (AB, AC and BC), on which the bilingual alignment is performed. The alignment set of the best scoring language pair (in his examples AB) is adopted and alignment is performed between this set and the sentences of the third language (C) using both remaining bilingual sets (the alignment sets of AC and BC).

The problems arising with non-conforming texts like the one shown in Table 4.2 are avoided by merging one-to-many and many-to-many alignments into single units after the first alignment. In that case, assuming that English and Spanish is the most similar language pair, the first two English and Spanish sentences and the third and fourth English sentence together with the third Spanish one will generate three joint sentences in the English+Spanish dimension:

$$A_{en+es} = \left[\binom{1}{1}, \binom{1}{1}, \binom{2}{1} \right]$$

The only way to combine the German sentences with these 'hyper sentences' (i.e., AUs) is a three-to-two alignment, that is, the three English+Spanish AUs with two German sentences. This yields a single all-embracing AU for the three languages:

$$A_{en+es+de} = \left[\binom{4}{3}{2} \right]$$

In cases like this, when a resulting trilingual AU comprises at least two sentences in at least two languages, Simard (1999) performs a bilingual sub-alignment of the AU in question, for which he reports a small performance improvement in terms of F-Score gain, measured on manually produced reference alignments.

In comparison to the raw bilingual alignment, he reports significantly better alignment results by his trilingual alignment method and concludes that aligning three languages "yields better bilingual alignments than can be obtained with bilingual text-alignment methods." We considered implementing a similar approach, but aligning 16 languages in the same way entails that AUs potentially grow larger with each subsequent combination of bilingual alignments, which consequently renders the task of sub-alignment ("the more challenging problem of finer-grained alignments" (ibid.)) more prominent. In the worst case, the resulting unified multilingual alignment sets (i.e., the transitive closure of all unified bilingual alignments) equals the entire sequences of sentences in a parallel text.

4.2.2 Evaluating Sentence Alignment

All published approaches (for an overview, see Tiedemann 2003b, Section 2.2; A. K. Singh and Husain 2005; Costa-jussà and Banchs 2011; Tiedemann 2011; Torres-Ramos and Garay-Quezada 2015) have one thing in common: They claim to be better than previous approaches, at least for particular use cases. Unfortunately, evaluation methods and data differ and, hence, it is inappropriate to compare raw values reported in those publications.

For a meaningful comparison of different aligners, the same set of evaluation texts needs to be aligned by these aligners and compared with the manually created gold alignment of the texts, using the same evaluation metric. Given that the quantity of texts is sufficient to attain significantly accurate values, the results depend, inter alia, on the text type, the evaluation metric and, if applicable, external language-specific resources provided to the aligners.

Regarding text type, two properties play a role: the quantity of null alignments and distribution of alignment types, that is, the numbers of corresponding sentences for each language in an AU.[15] Alignment approaches designed to deal with large amounts of non-translated sentences, such as (Braune and Fraser 2010), will excel when it comes to identifying null alignments. Alignment units that contain more than two sentences in one language are difficult to identify for most of the aforementioned approaches, but some of them are limited to particular numbers

[15] Krynicki (2012) uses the term *structural fidelity* to denominate the ratio of one-to-one alignments among all gold alignments. Yu et al. (2012) point out that literary texts are typically not translated sentence-wise, which leads to a higher amount of one-to-many and many-to-many alignments in literature.

by design. Gale and Church (1991) allow at most two sentences in each language, (Braune and Fraser 2010) requires AUs to comprise exactly one sentence in one of the languages.

Despite the above described difficulties in comparing sentence alignment methods, several comparisons have been conducted, each of which with one or more particular use cases in mind. Existing implementations of those alignment methods have typically been used when available, other methods have been reimplemented by the respective authors.

A. K. Singh and Husain (2005) evaluate four alignment methods for the language pair English/Hindi. They only evaluate one-to-one alignments for "practical constraints" and "because 1-to-1 alignments are the ones that can be most easily and directly used for linguistic analysis as well as machine learning." From our point of view, the latter argument remains disputable if not further qualified. Abdul-Rauf et al. (2012) evaluate five different alignment methods with regard to the results that the aligned sentences achieve when used for training statistical machine translation systems. Krynicki (2012) evaluates four alignment methods, which are also covered by the evaluation of Abdul-Rauf et al. (2012), on English/Polish parallel texts. Unlike the previous two evaluations, he discusses the pros and cons of different approaches and gives recommendations for the respective aligners in view of different applications, such as terminology extraction or lexicography.

Torres-Ramos and Garay-Quezada (2015) discuss the ideas behind and properties of several sentence alignment methods, some of them overlapping in concepts. They recapitulate the results reported by the respective publications and come to the conclusion that "[t]here is not one statistical-based approach that works for all kind of languages in the scope of parallel corpus alignment" and that "there is much to improve" with respect to external resource usage. In this regard they point to joint approaches using both "lexical and statistical information".

4.3 Multilingual Sentence Alignment

The example of Table 4.2 demonstrates that the task of multilingual sentence alignment is not trivial.[16] Even though we can achieve a low error rate by aligning most of the one-to-one correspondences correctly, the one-to-many and many-to-many alignments pose a challenge to any sentence alignment algorithm.

When it comes to identifying corresponding sentences in more than two languages, things get even more complex. A text with strictly parallel sentences like the one shown in Table 4.1 and Figure 4.2, that is, only one-to-one or rather one-to-one-to-one (1:1:1) alignments, represents the trivial case, where we only need

[16] Simard and Plamondon (1998) show that bilingual sentence alignment is not trivial either.

to include one more sentence in another language. This AS could be obtained by performing pairwise sentence alignment and then combining those bilingual ASs, which corresponds to the method suggested by (Simard 1999). If we choose the language pair German/English for the initial AS in Table 4.1, the ASs of both German/Spanish and English/Spanish agree on the final trilingual AS.

Table 4.2 and Figure 4.3 show an example where this naïve method for multilingual alignment will fail. All three languages have only beginning and end in common but no boundary of AUs in between. Combining two pairwise alignments (no matter which ones) would result in a single AU containing all the sentences shown, as the respective language pairs do not agree on any inner alignment boundary. In general, when we add parallel texts in other languages, the chance of disagreement increases and the AUs become potential bigger with each additional language.

In addition to the AUs becoming unhelpful when they grow too big, another problem of combining pairwise AUs to multilingual alignments is consistency. It is not guaranteed that combining the ASs of, for instance, German with English and subsequently English with Spanish will result in the same set obtained by aligning German with Spanish (as it is the case in Table 4.1). Again, the more languages and thus language pairs involved, the higher the probability of disagreement between the respective ASs. An alignment approach based on pairwise alignment thus needs a way to handle disagreements in order to generate consistent multilingual alignments. Simard (ibid.) solves this problem for trilingual alignment by summing up bilingual alignment scores of all sentence pairs that belong to each potential trilingual AU.

To benefit from those presumably big but consistent multilingual alignments, we need to also indicate contained AUs of less than the total number of languages that the former includes.[17] In the example in Table 4.2, we would want our multilingual AU, which comprises the whole list of sentences in all three languages, to contain five smaller AUs.[18] The data structure required for holding those AUs requires partially overlapping AUs (take, for instance, the first English sentence), which would make those multi-level multilingual alignments difficult to handle. In order to limit complexity, we resort to **hierarchical alignments**, which require that for any two AUs that have at least one element (here: a sentence) in common, one is a proper subset of the other (see Graën and Clematide 2015).

Hierarchical alignments are, from a formal point of view, tree structures such that the **leaf nodes** are AUs, which comprise a set of sentences (in general: elements), and **branch nodes** include a set of AUs, thus (indirectly) also comprising their elements. The topmost node, the **root node**, which can also be a leaf node

[17] Simard (1999) refers to it as "the more challenging problem of finer-grained alignments" where he expects to "encounter numerous complications."

[18] Namely for English/Spanish the sentence pairs [(1), (1)], [(2), (2)] and [(3,4), (3–4)] and for English/German the sentence pairs [(1,2,3), (1–3)] and [(4), (4)].

Table 4.4 – Excerpt from a gold-aligned text in 16 languages.

	Sentence	AUs	
bg_1	Тук бизнесът не играе изместваща роля, а допълваща, и решаващият момент е, че изследванията ще останат свободни, точно както преподаването.	1	
de_1	Dabei hat die Wirtschaft keine übernehmende, sondern eine ergänzende Funktion, und entscheidend ist, dass die Forschung frei bleibt und die Lehre ebenso.		
el_1	Ο ρόλος των επιχειρήσεων εν προκειμένω δεν είναι εκτοπιστικός αλλά συμπληρωματικός, και το σημαντικό στοιχείο είναι ότι η έρευνα θα παραμείνει ελεύθερη όπως ακριβώς και η διδασκαλία.		
en_1	Business does not have a supplanting role here, but a complementary one, and the crucial point is that research will remain free just as teaching will.		
es_1	Las empresas no actúan como sustitutas aquí, sino que adoptan un papel complementario, y el punto fundamental es que la investigación se mantenga libre, al igual que la enseñanza.		
ro_1	Mediul de afaceri nu are un rol supleant, ci unul complementar, iar aspectul esenţial este faptul că cercetarea va rămâne independentă, ca şi predarea.		
en_2	It will make its own decision in this regard;	2	6
es_2	Tomará su propia decisión a este respecto;		
ro_2	Va lua propriile sale decizii în ceea ce o priveşte;		
en_3	it will not be forced into this by politicians.	3	
es_3	los políticos no la obligarán.		
ro_3	politicienii nu o vor obliga nicicum în acest sens.		
bg_2	В това отношение решенията ще се вземат от самите изследователи и преподаватели, а няма да бъдат налагани от политиците.	4	
de_2	Sie entscheidet sich dafür, die Politik zwingt sie nicht.		
el_2	Θα λαμβάνει τις δικές της αποφάσεις από την άποψη αυτή, χωρίς να της ασκούνται πιέσεις από τους πολιτικούς.		
bg_3	Имаме нужда от печеливша за всички ситуация, което означава науката и образователните институции, от една страна, и изследователите и бизнесът, от друга, да кажат „да“ на това партньорство.	5	
de_3	Wir brauchen eine Win-Win-Situation, in der die Organe der Wissenschaft und Lehre einerseits und der Forschung und Wirtschaft anderseits Ja zu dieser Partnerschaft sagen.		
el_3	Χρειαζόμαστε ένα εξασφαλισμένο αποτέλεσμα, σύμφωνα με το οποίο δηλαδή τα επιστημονικά και εκπαιδευτικά ιδρύματα, αφενός, και η έρευνα και οι επιχειρήσεις, αφετέρου, θα πουν «ναι» σε αυτήν την εταιρική σχέση.		
en_4	We need a win-win situation, that is, one in which science and teaching institutions, on the one hand, and research and business, on the other, will say ‘yes’ to this partnership.		
es_4	Necesitamos una situación beneficiosa para todos, es decir, una en la que la ciencia y las instituciones de enseñanza, por una parte, y la investigación y la empresa, por la otra, digan «sí» a esta asociación.		
ro_4	Avem nevoie de o situaţie în care toată lumea să câştige, adică o situaţie în care ştiinţa şi instituţiile de educaţie, pe de-o parte, şi cercetarea şi mediul de afaceri, de cealaltă parte, să spună „da” acestui parteneriat.		

in case there is no superordinate AU, is the minimal AU necessary to cover all particular languages. It can correspond to the entire text alignment if the respective languages do not agree on an intermediate alignment boundary. If this is not the case, the AS of a text is a forest of hierarchical AUs. We show an example in Table 4.4.[19]

We model hierarchical AUs as sets such that a superordinate AU comprises all elements of a subordinate AU. The condition that two AUs, A_1 and A_2, from the same alignment set $\mathcal{A}$ need to be distinct with regard to their elements if one does not comprise the other is expressed by the following equation:

$$\forall(A_1, A_2) \in \mathcal{A} \times \mathcal{A} : A_1 \subset A_2 \vee A_1 \supset A_2 \vee A_1 \cap A_2 = \emptyset \tag{4.1}$$

While the AUs in a hierarchical AS are sets of sentences, we additionally require the sentences of each AU to be in monotonic order, thus not allowing for "crossing" AUs or intermediate sentences to belong to a different AU. Unlike sentence alignment for "asymmetric parallel corpora" (see Braune and Fraser 2010), the translators of the European Parliament's debates do typically not omit whole sentences (omitted or additional words, such as in Table 4.3, can be found, though). The following equation specifies the requirement that if one sentence (s_1) precedes ($<$) another one (s_2), the same must hold for all sentences in the respective AUs A_1 and A_2:

$$\begin{aligned} &\forall(A_1, A_2) \in \mathcal{A} \times \mathcal{A} : \forall(s_1, s_2) \in A_1 \times A_2 : \\ &s_1 < s_2 \implies \forall(s_x, s_y) \in A_1 \times A_2 : s_x < s_y \end{aligned} \tag{4.2}$$

Having a hierarchical alignment set (hierarchical AS), we can obtain the relevant AUs for any subset of the languages comprised by the AS, the **minimal alignment set** for those languages, by removing the elements of all other languages from the AUs and subsequently removing any resulting duplicate AU. In cases like the one shown in Table 4.2 where sentences overlap without being contained by one another, we can either represent the smaller AUs of English/German or English/Spanish in a hierarchical AS. Such a constellation is rather infrequent as we will see in Section 4.3.2. The more frequent case that we find in our corpus texts is that some languages use subordinate clauses while other languages prefer two single sentences. This is depicted in Table 4.5.

[19]For our multilingual sentence alignment approach described in Section 4.3.1, we limit the number of hierarchical levels to two, that is, we only allow for one level of branch nodes (which are also root nodes) on top of the leaf nodes. This limitation is motivated by our experience with manual sentence alignment; a third level of AUs could have been utilized only in a very limited number of cases.

Table 4.5 – Two sentences versus one sentence with a subordinate clause. The original language is French in both examples.

	English	German	Spanish
1	You see before you a Parliament of elected representatives who, each time they meet their constituents, have to justify the collective impotence of the Member States and of the Union when it comes to unemployment, which is becoming more and more of a scourge.	Sie sehen ein Parlament Volksvertreter vor sich, die sich bei jeder Begegnung mit ihren Wählern für die allgemeine Unfähigkeit unserer Staaten und der Union, die Arbeitslosigkeit zu bekämpfen, rechtfertigen müssen.	Tiene usted ante sí un Parlamento de representantes que, siempre que se reúnen con sus electores, deben justificar la impotencia colectiva tanto de nuestros Estados como de la Unión en materia de desempleo.
2		Mit dieser Plage wird es immer schlimmer.	La gravedad de ese flagelo aumenta constantemente.

	English	German	Spanish
1	If shipbuilding is necessary for Europe's economy, it must be brought under state control.	Wenn der Schiffbau ein wichtiger Bestandteil der europäischen Wirtschaft ist, dann muss er verstaatlicht werden, anstatt die Privateigentümer mit nichtrückzahlbaren Subventionen zu überhäufen.	Si la construcción naval es necesaria para la economía europea, hay que nacionalizarla, en lugar de subvencionar a fondo perdido a sus propietarios privados.
2	The answer is not to subsidise its private owners with money that we will never see again.		

Apart from the objective of identifying multilingual AUs for the sake of investigating linguistic phenomena by comparison of more than two languages, we also expect that supplemental evidence in the form of additional languages will increase accuracy of bilingual alignments, which corresponds to the minimal AS for a given language pair. Triangulation approaches, that is, the use of a third language to deduce or improve relations for a given language pair, have successfully been applied to word alignment (see, for instance, Cohn and Lapata 2007).

4.3.1 Our Approach to Multilingual Sentence Alignment

Our alignment algorithm proceeds in four steps. First, an undirected graph is built with the respective sentences as nodes and alignment scores as edges. For every two nodes from distinct languages and each feature (see below), we calculate the alignment score – based on partial values for different alignment features – and, if

appropriate (i.e., the alignment score is above a given threshold value), establish an edge between these two nodes. A pair of unconnected nodes is defined to have an alignment score of 0.

The second step performs single-linkage hierarchical agglomerative clustering on the derived graph such that every resulting cluster comprises at most one sentence of each language. In the following third step, incomplete clusters, that is, those that do not possess sentences in all languages, are clustered with neighboring complete clusters.[20]

The last step is the conversion of the resulting cluster structure into multilingual alignments, which – given the assumption that a sentence in one language needs to have translations in all other languages – entails the separation of those sentences from the complete cluster into a separate one that also forms part of any second level cluster.

Features

Some bilingual features that we use for multilingual sentence alignment need parallel sentences to learn from. We obtain appropriate sentence pairs for each pair of languages using a bootstrapping approach: First, we find all parallel texts of a given language pair that have the same number of sentences according to our sentence segmentation. Second, we calculate the token ratio of all texts of that language pair and rank them according to their relative deviation from the overall token ratio for that pair.[21] From this ranking, we only use parallel texts with a token ratio similar to the expected one, that is, the ratio we get from the absolute numbers of tokens available in a subcorpus of texts in these two languages. The intersection of both lists is our selection of superficially well-fitting parallel texts, which we expect to hold a straight one-to-one correspondence of sentences.

The number of – supposedly – parallel sentences in the well-fitting parallel texts ranges from 27 223 for Bulgarian/Polish (4664 texts) to 283 189 for English/Slovene (31 262 texts). These texts cover 14.1 % and 71.4 %, respectively, of parallel texts available for both pairs in total. The percentage of well-fitting parallel texts per language pair depends on the number of texts available in each language, which is not necessarily equal to but approximately the lower number of texts available in both languages (see Figure 4.1) and the fit calculated above.

We use a set of twelve features of the form $\phi_y(s^1, s^2)$ for constructing edges (alignment indicators) between nodes (sentences) of different languages (s^1 and s^2).

[20] These two steps are similar to what Braune and Fraser (2010) do but with more than two languages.

[21] We use the token ratio and not character ratio, which Gale and Church (1991) found to give better results for sentence alignment, as we intend to retrieve lexical information from those sentence pairs.

Each feature is allowed to take numeric values from the interval $[0, 1]$, indicating a gradient degree of evidence for correspondence. If a feature's formula returns values outside this range, they are mapped to 0 (negative values) or 1 (positive values). The respective features target general lexical correspondence (ϕ_{pt} and ϕ_{pl}), matching numbers and acronyms (ϕ_{no} and ϕ_{ac}), discourse markers (ϕ_{ft} and ϕ_{fl}), punctuation (ϕ_{ia} and ϕ_{iq}), positional and length information (ϕ_{l1}, ϕ_{l2}), and we use alignment scores from bilingual sentence alignment of each language pair (ϕ_{et} and ϕ_{el}). The following list elaborates on feature design and calculation:

- ϕ_{pt} and ϕ_{pl} are based on phrase table matches for word forms and lemmas, respectively. We extract those phrase tables for each language pair from the set of well-fitting parallel texts using *anymalign* (Lardilleux and Lepage 2009). Anymalign is designed to retrieve words and phrases from parallel sentences in multiple languages (we only use it for language pairs) by comparing distributions of word forms and sequences of word forms (see also page 115). Word forms without a corresponding lemma are excluded from the input sentences fed to anymalign.

  ϕ_{pt} (and likewise ϕ_{pl}) for a pair of sentences (s^1, s^2) is defined as

  $$\phi_{pt} = \sum_{(T^1,T^2)\in\mathcal{P}(s^1)\times\mathcal{P}(s^2)} ln\left(1 + |T^1| \cdot |T^2| \cdot p_w(T^2|T^1) \cdot p_w(T^1|T^2)\right) \quad (4.3)$$

  where $\mathcal{P}$ is the power set, that is, a set containing all possible subsets of tokens, and p_w is the lexical weight (of sequences of either word forms or lemmas) as defined by (ibid.).[22] The lexical weight of any combination not comprised by the respective phrase tables (including noncontinuous combination) defaults to 0; hence, those combinations do not contribute to the sum.

  For the common phrases "Разискването приключи" and "συζήτηση έληξε" *'the debate is over'* in the Bulgarian/Greek word form phrase tables, anymalign reports p_w to be 0.6836 and 0.9548, respectively. As both phrases consist of two tokens each, the numerator of the inner fraction evaluates to 1.2839. In theory, this formula permits values higher than 1.0; in practice, we do predominantly see low values of ϕ_{pt} and ϕ_{pl}.

- ϕ_{ft} and ϕ_{fl} are based on matching first tokens. These features are similar to the phrase-based features ϕ_{pt} and ϕ_{pl}, but restricted to the first token of each sentence. Our motivation in building an additional feature just for the first

[22] Their notion of the lexical weights differs from Koehn, Och et al.'s (2003) original definition insofar as they use the maximal lexical translation probability between a word in one language and a sequence of words in the other instead of averaging the respective lexical translation probabilities.

token of each sentence was to capture potential discourse markers. Given that all parallel texts, original or translations, convey the same argument structure, we expect to frequently encounter corresponding discourse markers on the first position in parallel sentences. These features are also calculated on our sample of well-fitting parallel texts.

ϕ_{ft} (and likewise ϕ_{fl}) is defined as

$$\phi_{ft} = \frac{min(p(s_1^1|s_1^2), p(s_1^2|s_1^1))}{\sqrt{f(s_1^1)} \cdot \sqrt{f(s_1^2)}} \tag{4.4}$$

with s_1^1 and s_1^2 being the first tokens of the respective sentences s^1 and s^2 and $f(t)$ the frequency of a particular word form or lemma in the text that is to be sentence-aligned. The feature value of initial tokens whose attributes (word forms or lemmas) are observed with higher frequencies in the text is reduced by the denominator.

In the Finnish/Polish list of initial word forms, 'Vaikka' and 'Chociaż' *'although/though'* appear together 33 times. While 'Chociaż' is translated in all 33 cases with 'Vaikka', these 33 cases only make up a 49.3 % of Finnish sentences that start with 'Vaikka'. For a parallel text where both words appear once, ϕ_{ft} thus equals to 0.493. If the pair appears twice in the text, ϕ_{ft} drops to 0.246.

- ϕ_{ia} and ϕ_{iq} are based on matching punctuation marks. While ϕ_{ia} applies to any punctuation,[23] ϕ_{iq} is only used for matching question marks. The motivation for introducing a separate feature for question marks is that they are typically used to indicate a question[24] and questions are typically translated as questions, while periods and exclamation marks, for instance, are interchangeable to some extent.

  To map the appearance of equal punctuation marks to a numeric feature, we take the relative frequencies of that mark with regard to the number of sentences for each particular language, multiply both results and project them, again, into the numeric interval $[0, 1]$ so that a punctuation mark that appears only once in both languages (e.g., a quotation mark) receives

[23] Including combined punctuation marks as in "Zij willen niet?!" *"They won't?!"*

[24] Except for Greek, where the semicolon is used instead (e.g., "Λειτουργεί σήμερα αποτελεσματικά η Ευρωπαϊκή Ένωση;" *"Is the European Union functioning efficiently today?"*). Question marks are used in Greek texts in Europarl, though, but predominantly in quotations: "Επιτρέψτε μου να ρωτήσω: Quousque tandem – Noiz arte?" *"May I ask: Quousque tandem – Noiz arte?"* In plenary debates up to January 1998, accented letters are frequently replaced with a question mark in the original Europarl corpus (e.g., 'Ευρ?πη' instead of 'Ευρώπη') (see also Section 2.3.1).

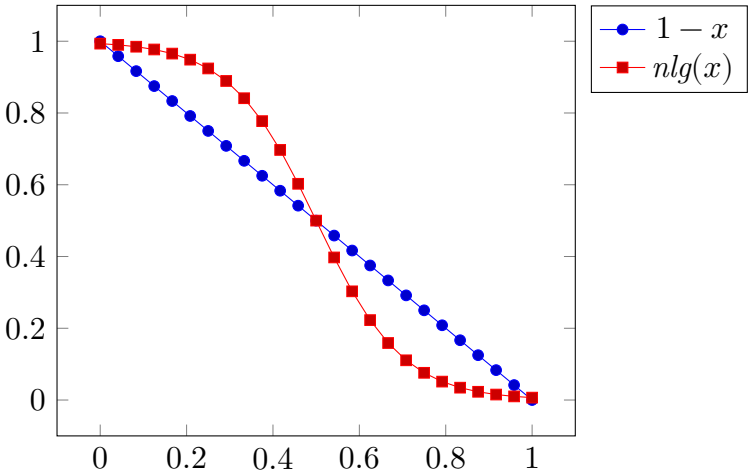


Figure 4.4 – Functions to map low values to high ones and vice versa (within the interval from 0 to 1). Unlike the linear transformation (blue), the negative logistic growth function (red) does not equally distribute input values from $[0, 1]$ to $[0, 1]$, but concentrates them towards both ends.

a higher value compared to one that appears frequently (e.g., a period). We use the negative logistic growth function *nlg* for mapping as it allows for better discrimination between frequent and infrequent marks:[25]

$$nlg(x) = 1 - \frac{1}{1 + e^{-10(x-0.5)}} \tag{4.5}$$

Figure 4.4 contrasts the *nlg* function with a linear negative projection. The feature ϕ_{ia} (and likewise ϕ_{iq} for question marks only) is defined as

$$\phi_{ia} = nlg(p(s^1_{|s^1|}) \cdot p(s^2_{|s^2|})) \tag{4.6}$$

with p being the relative frequency of the last token's surface form in all sentences of the language in question.

For a short parallel text with ten sentences in each language and two of them ending with an exclamation marks, one with a question mark and seven with periods, we obtain 0.993 as result for the two sentences with question marks, 0.990 for each combination of the four sentences with exclamation marks and 0.198 for any combination of sentences with a regular period, thus favoring

[25] Any other sigmoidal function should give comparable results. The linear function, however, performs worse.

the question and exclamation marks over the more frequent periods. An additional edge is established by ϕ_{iq}, which only applies to sentences ending with question marks.

It is typically the period that is predominantly used in a text to end sentences and these sentences consequently receive ϕ_{ia} values close to 0. However, in speaker contributions where rhetorical questions prevail, it is, for instance, the infrequent period that receives a high ϕ_{ia} value, while the sentences ending with question mark score considerably lower. The ϕ_{iq} values will be accordingly low in such a case.

- ϕ_{l1} and ϕ_{l2} are length-based features. While ϕ_{l1} implements a linear model, ϕ_{l2} makes use of the negative logistic growth function to penalize higher deviation in lengths.

  We measure the length of a sentence s in terms of characters of its tokens ($len(s) = \sum_{t \in s} len(t)$), disregarding any potential space characters between them. The relative cumulative length for the nth sentence in a text x (i.e., x_n) is the length of all preceding sentences plus the length of the nth sentence itself divided by the total length of all sentences in that text:

$$\delta_n(x) = \frac{\sum_{i=1}^{n} len(x_i)}{\sum_{i=1}^{|x|} len(x_i)} \tag{4.7}$$

  It thus describes the position of the right sentence boundary of the sentence in question in relation to the whole text. The left sentence boundary corresponds to the right boundary of the preceding sentence (i.e., δ_{n-1}). By definition, the relative cumulative length of the first sentence's left boundary (i.e., δ_0) always equals 0, and the relative cumulative length of the last sentence's right boundary (i.e., $\delta_{|x|}$) always equals 1.

  The relative cumulative length difference ($rcld$) of two alignment boundaries i and j in texts x^1 and x^2 is an indicator of how well those boundaries fit together length-wise:[26]

$$rcld_{i,j}(x^1, x^2) = \left|\delta_i(x^1) - \delta_j(x^2)\right| \tag{4.8}$$

[26] For "asymmetric parallel corpora" (Braune and Fraser 2010), this measure will be less useful than for more faithful translations such as parliament proceedings.

n	Slovak	δ_n
1	Cieľom správy o revízii smernice o odpade z elektrických a elektronických zariadení bolo podporiť separovaný zber, zhodnocovanie a recykláciu tohto druhu odpadu.	34 %
2	Teoreticky by som si želal podporiť tento prístup.	45 %
3	K správe sa však pridalo množstvo pozmeňujúcich a doplňujúcich návrhov, ktoré predstavujú záťaž pre drobných obchodníkov.	70 %
4	Tí musia znášať ďalšie administratívne náklady a plniť ďalšie požiadavky, čo im spôsobí problémy.	91 %
5	Preto som sa rozhodol hlasovať proti návrhu.	100 %

n	Swedish	δ_n
1	Syftet med betänkandet om översyn av direktivet om avfall som utgörs av eller innehåller elektriska eller elektroniska produkter är att stimulera till separat insamling, utvinning och återanvändning av den här typen av avfall.	44 %
2	I teorin hade jag velat stödja den här strategin, men det har tillkommit ett antal ändringsförslag som är betungande i synnerhet för små affärsinnehavare, som påtvingas administrativa kostnader och krav som är svåra att klara.	88 %
3	Av den orsaken bestämde jag mig för att rösta emot förslaget.	100 %

		Slovak				
		1	2	3	4	5
Swedish	1	0.99	0.99	0.92	0.59	0.36
	2	0.64	0.98	0.99	0.99	0.98
	3	0.17	0.40	0.88	0.98	0.99

Figure 4.5 – ϕ_{l2} values for a short parallel text in Slovak and Swedish.

The trivial cases of the first sentences' left and the last sentences' right boundaries, $rcld_{0,0}(x^1, x^2)$ and $rcld_{|x^1|,|x^2|}(x^1, x^2)$, which are aligned by definition, consequently both yield a relative cumulative length difference of 0. For the feature ϕ_{l2} applied to a sentence pair (x_i^1, x_j^2), we use the lesser relative cumulative length difference of both corresponding sentence boundaries, which in turn gives a higher value when the *nlg* function is applied to it:

$$\phi_{l2} = nlg(min(rcld_{i-1,j-1}, rcld_{i,j})) \tag{4.9}$$

An example for this feature is shown in Figure 4.5. For the first and the last sentences of parallel texts, ϕ_{l2} will accordingly be close to 1 as *rcld* yields 0 for these sentences.

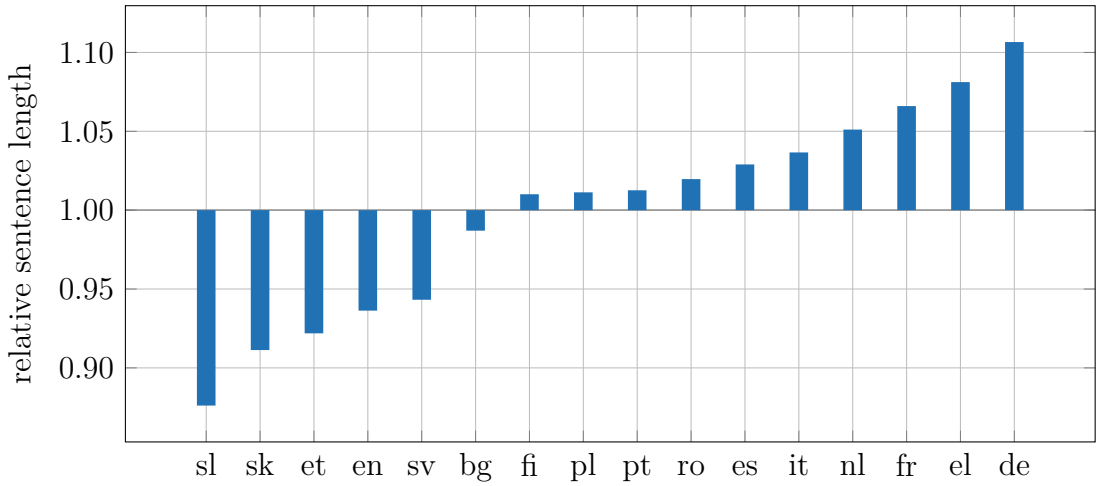


Figure 4.6 – Relative sentence lengths (Δ) in terms of characters. The difference averages out at 1.0 over all 16 languages.

For ϕ_{l1}, we first calculate the length ratio of both texts. To this end, we divide the shorter by the longer one, also taking into account the relative length differences inherent to the set of languages that we work with. Figure 4.6 shows the relative lengths in terms of characters, which we refer to as Δ.[27] We obtained these values by counting characters on the well-fitting parallel texts that are available in all languages and dividing the respective counts by the global average. Comparing the extremes, we expect a German text to use approximately 25 % more characters than a parallel Slovene one.

The ratio of normalized lengths of two sentences s^1 and s^2 is a simple measure of length-based fit:

$$r = \frac{min\left(\sum_i len(s_i^1)/\Delta^1, \sum_i len(s_i^2)/\Delta^2\right)}{max\left(\sum_i len(s_i^1)/\Delta^1, \sum_i len(s_i^2)/\Delta^2\right)} \tag{4.10}$$

A sentence pair with corresponding lengths (e.g., a Slovene sentence with 100 and a German sentence with 125 characters) yields values close to 1. The greater the difference of the normalized length values, the lower the value of r will be.

[27] Kay and Röscheisen (1993) report concordantly that "one character in English on average gives rise to somewhat more than 1.2 characters in German." Gale and Church (1991) found a ratio of 1.1 for English/German and 1.06 for English/French.

Based on r, the relative length of the right sentence boundary and another factor representing the length difference of both sentences, we define ϕ_{l1} as:

$$\phi_{l1} = r \cdot (1 - rcld_{i,j}) + 0.5 \cdot \left(1 - \frac{\left|len(s^1)/\Delta^1 - len(s^2)/\Delta^2\right|}{max\left(len(s^1)/\Delta^1, len(s^2)/\Delta^2\right)}\right) \quad (4.11)$$

Both features ϕ_{l1} and ϕ_{l2} make use of the relative cumulative length difference. In contrast to ϕ_{l2}, ϕ_{l1} also takes into account the relative lengths of the sentence pair in question. Our assumption motivating two different features based on sentence lengths is that the better feature will prevail at feature weight optimization (see below); we expect the other one to receive a low weight accordingly.

- ϕ_{no} rewards matching numbers. Longer matching numbers in terms of digits receive a higher value. ϕ_{ac} does the same for acronyms. Numbers and acronyms are also extracted from tokens if their word form contains at least one numeral or two subsequent uppercase letters.[28] That way, we are also able to match, for instance, tokens in languages that typically add case endings to both kinds of tokens (e.g., English 'the 20th century', German 'das 20. Jahrhundert', French 'le XXe siècle', Spanish 'el siglo XX').

  The values for both features are calculated as

$$\phi_{ac} = \prod_{(t^1,t^2)\in s^1\times s^2} nlg\left(\frac{\sqrt{f(t^1)}\cdot\sqrt{f(t^2)}}{len(t)}\right)\cdot\varrho(t^1,t^2) \quad (4.12)$$

  with ϱ being a function that yields 1 for matching acronyms and 0 in all other cases (acr returns all uppercase letters if a token's word form has at least two consecutive ones):

$$\varrho(t^1,t^2)\begin{cases}1 & \text{if } acr(t^1) = acr(t^2) \neq \emptyset \\ 0 & \text{otherwise}\end{cases}$$

  As the matching of numbers starts with a single digit (in contrast to at least two letters for acronyms), we increment the denominator of the fraction for ϕ_{no} by 1.

[28] Although Simard, Foster et al.'s (1993) measures slightly differ from ours, ϕ_{no} and ϕ_{ac} (and also ϕ_{ia} and ϕ_{iq}) can be subsumed as cognates.

- ϕ_{et} and ϕ_{el} are features for alignment externally generated by hunalign (Varga et al. 2005) on word forms and lemmas, respectively.[29] Hunalign provides a *confidence value* (c) for each alignment unit (AU) and an *alignment quality* (q) value per alignment set (AS).[30]

  All sentence pairs that form part of an AU identified by hunalign receive that AU's confidence value. The alignment quality value is invariant for the whole AS. Alignment on different language pairs, however, will in most cases result in different values and, since the respective links between all languages compete with each other, ASs with a higher alignment quality value will benefit from it.

  We combine both values, which are typically greater than 1, such that high values from hunalign are mapped to high values of ϕ_{et} (and likewise ϕ_{el}):

$$\phi_{et} = 1 - \frac{1}{1 + max(c(s^1, s^2), 0) \cdot max(q(s^1, s^2), 0)} \tag{4.13}$$

  Both of hunalign's ratings can take negative values as well. That is why we explicitly set 0 as lower limit for both values. If either c or q is negative, ϕ_{et} will yield 0, that is, no evidence from pairwise sentence alignment for the sentence pair in question is available.

Weights

The features listed above are calculated for all edges between sentence nodes of different languages. We use a linear combination of them to get a single alignment score per edge. To this end, we employ feature-specific weights w reflecting the importance of each feature. The final alignment score (as) is calculated as

$$as(s^1, s^2) = \sum_i w_i \cdot \phi_i(s^1, s^2) \tag{4.14}$$

We use a sampling algorithm to determine optimal weights for multilingual sentence alignment in our corpus (see Section 4.3.2). Other corpora with different characteristics (e.g., many null alignments) will most likely benefit from a different configuration of weights – or potentially require completely different features.

[29] We use hunalign since it is easy to apply and shows a good performance, in particular when it is equipped with corpus-specific dictionaries (see Section 4.3.2). Other sentence alignment tools require more resources. For bleualign (Sennrich and Volk 2010), for instance, we would need to train or acquire machine translation models for 120 language pairs.

[30] See documentation on https://github.com/danielvarga/hunalign.

An additional weight w_{ls}, which is used to reward coherent lengths of joined clusters in the secondary clustering step (see below), forms part of the optimization process (see Section 4.3.2) as we expect it to interact with the other features' weighting.

Primary Clustering Step

Our algorithm consists of three steps: First, we perform single-linkage hierarchical agglomerative clustering on the graph that is composed of sentences as nodes and alignment scores as edges. In doing so, we only permit monotonic clusters (see Equation 4.2) with a single sentence in each language. The second step is concerned with joining neighboring clusters, which leads to new clusters with a theoretically arbitrary number of sentences in each language. In practice, we predominantly see clusters that result from a single join, thus having at most two sentences in each language. In the third step, we convert the resulting clusters into hierarchical alignment sets.

The primary clustering step yields clusters that contain at most one sentence per language. It traverses the ordered list of precalculated alignment scores for sentence pairs starting with the highest one. Depending on the current state, one of the following three actions is taken:

1. Create a new cluster with both sentences if none of them forms part of an existing cluster.

2. Join an existing cluster if one of the two sentences forms part of it, the other one has not been assigned yet and there is no sentence of the same language present in the cluster.

3. Do nothing if both sentences form part of either the same (skip) or two distinct clusters (reject).

Any cluster constructed in this way either comprises one sentence in each language, that is, it is complete, or it is lacking sentences in at least one language, that is, it is incomplete. All sentences without association to a cluster generated in this step are assigned their own (necessarily incomplete) cluster so that, when this step is completed, every sentence is assigned to exactly one cluster.

Secondary Clustering Step

The secondary clustering step processes each incomplete cluster C_i and identifies complete candidate clusters C_c with which to join the incomplete cluster. The hypothetical joined clusters are required to not violate the requirement of monotonicity, that is, to not yield alignment units crossing any of the existing clusters.

Table 4.6 – Revisited example from Table 4.5. The colored part corresponds to the primary cluster (C_1), the non-colored part to the secondary one (C_2).

	English	German	Spanish
1	You see before you a Parliament of elected representatives who, each time they meet their constituents, have to justify the collective impotence of the Member States and of the Union when it comes to unemployment, which is becoming more and more of a scourge.	Sie sehen ein Parlament Volksvertreter vor sich, die sich bei jeder Begegnung mit ihren Wählern für die allgemeine Unfähigkeit unserer Staaten und der Union, die Arbeitslosigkeit zu bekämpfen, rechtfertigen müssen.	Tiene usted ante sí un Parlamento de representantes que, siempre que se reúnen con sus electores, deben justificar la impotencia colectiva tanto de nuestros Estados como de la Unión en materia de desempleo.
2		Mit dieser Plage wird es immer schlimmer.	La gravedad de ese flagelo aumenta constantemente.

Typically, there are two alternatives: the complete cluster following or preceding the incomplete one. The latter is the case in the first example in Table 4.5, revisited in Table 4.6, where the first, colored sentences make up a complete cluster ($C_1 = [(1), (1), (1)]$) and the two non-colored sentences form an incomplete cluster since an English sentence is missing ($C_2 = [\emptyset, (2), (2)]$).

In case an incomplete cluster has two neighboring complete ones, both candidates are ranked according to the alignment scores between their components:

$$cas(C_i, C_c) = \sum_{(s_i, s_c) \in C_i \times C_c} as(s_i, s_c) \tag{4.15}$$

The feature weight w_{ls} is then added to the cluster alignment score (cas) of that candidate C_c that, joined with C_i, would be more coherent, that is, the one with the lowest root-mean-square deviation of normalized lengths. In so doing, we account for the strong sentence alignment feature of length correspondence (see Section 4.2.1), which we considered less helpful for the primary clustering step, whose objective is not to find complete sentence correspondences but the nuclei of potentially larger AUs. We join the candidate with the highest total score with C_i and consider the resulting joined cluster as complete for subsequent actions.

As there may be clusters that cannot be joined to any complete cluster at first, the process of joining is repeated until no further joins can be performed. The clustering process allows for persisting incomplete clusters, although we expect every sentence in our corpus to have a counterpart in all other languages, be it a single sentence, a sequence of sentences or parts of a sentence. Fortunately, persisting incomplete clusters are not a frequent phenomenon according to our observations.

In the example from Table 4.6, the second clustering step will join the first, complete cluster with the second, incomplete cluster resulting in a single cluster: $C_{1+2} = C_1 \cup C_2 = [(1), (1, 2), (1, 2)]$

Transformation into Hierarchical Alignment Sets

The two clustering steps above aim at identifying the best matching single sentences (primary clustering) and all other sentences that also belong to those best matching sentences (secondary clusters) in case the correspondence is not trivial and all alignment boundaries agree (as is the case in Table 4.1).

To convert the obtained cluster structure into hierarchical ASs, we first separate the sentences of those languages that are present in any attached incomplete cluster from the complete one into a new cluster and, second, let the attached incomplete clusters, the new cluster and the remaining sentences from the primary cluster join on a higher level. For the first example from Table 4.6 with the first sentences in all languages forming the complete ($C_1 = [(1), (1), (1)]$) and the second sentences in German and Spanish ($C_2 = [\emptyset, (2), (2)]$) forming the incomplete cluster, this means moving the German and Spanish sentences (i.e., the languages that are also present in the incomplete cluster), from the complete into a new cluster ($C_3 = [\emptyset, (1), (1)]$) and removing them from C_1 ($C'_1 = [(1), \emptyset, \emptyset]$). The final step is to join all three resulting clusters to a superordinate one ($C_4 = C'_1 \cup C_2 \cup C_3 = [(1), (1, 2), (1, 2)]$.

The top-level cluster C_4 thus comprises all example sentences and contains two sub-clusters, namely the respective first (C_3) and second (C_2) sentences in German and Spanish. Cluster C'_1 is disregarded as it is contained in C_4. Furthermore, it only comprises one language and, hence, does not align anything. Table 4.7 depicts this structure.

4.3.2 Evaluation

To evaluate our multilingual alignment approach, we use two methods: one that evaluates the generated multilingual ASs with respect to the performance on its included minimal bilingual ASs for language pairs; and one that focuses on **multilingual alignment consistency**, accounting for all languages included in the alignment process.

As a foundation with which to compare our automatically obtained ASs, we manually sentence-aligned 100 randomly chosen texts (i.e., documents), our **gold alignments**. Minimal ASs for language pairs can be extracted as described in Section 4.3 on page 78.

Table 4.7 – The resulting clusters that constitute the hierarchical alignment set (except for cluster C_1^1, which only contains a single sentence in one language).

	Sentence	AUs	
en_1	You see before you a Parliament of elected representatives who, each time they meet their constituents, have to justify the collective impotence of the Member States and of the Union when it comes to unemployment, which is becoming more and more of a scourge.	C_1'	C_4
de_1	Sie sehen ein Parlament Volksvertreter vor sich, die sich bei jeder Begegnung mit ihren Wählern für die allgemeine Unfähigkeit unserer Staaten und der Union, die Arbeitslosigkeit zu bekämpfen, rechtfertigen müssen.	C_3	
es_1	Tiene usted ante sí un Parlamento de representantes que, siempre que se reúnen con sus electores, deben justificar la impotencia colectiva tanto de nuestros Estados como de la Unión en materia de desempleo.		
de_2	Mit dieser Plage wird es immer schlimmer.	C_2	
es_2	La gravedad de ese flagelo aumenta constantemente.		

Gold Alignments

To evaluate our alignment approach, we manually aligned sentences in 100 texts from FEP9 in up to 16 languages using the Hierarchical Alignment Tool (HAT) shown in Appendix A.2. Six of the originally selected 100 texts had to be replaced because they were not parallel, that is, they did not comprise mutual translations for all languages, or at least one sentence boundary was not recognized, for instance, due to use of a centered dot (i.e., an interpunct: ‘ · ’) instead of a regular one. We found three cases where the period of an ordinal number in Polish was misinterpreted by our tokenizer as a cardinal number followed by a sentence-final period, which led to the recognition of incorrect sentence boundaries. Since we expect the alignment algorithm to later merge the resulting fragments, we decided to continue keeping those incorrectly split sentences.

Not all texts are available in every language. We count 62 texts comprising all 16 languages, 5 with 15, 27 with 14 and 6 with 13 languages. Altogether, the 100 selected texts account for 14 892 sentences; the longest text consists of approximately 80 sentences in each language and there are only three texts with more than 40 sentences (see Figure 4.7). On average, each text comprises 9.8 sentences with the most prominent deviations being Romanian with 8.4 sentences and Polish with 10.6 sentences. These numbers are also depicted in Figure 4.7.

We limit the alignment hierarchy to two levels to generate a structure equivalent to the one produced by our alignment algorithm and, first and foremost, to not further prolongate the time-consuming process of manual alignment. Sentences aligned in leaf nodes can thus be joined to branch nodes once, but joining branch nodes in larger structures is not supported.

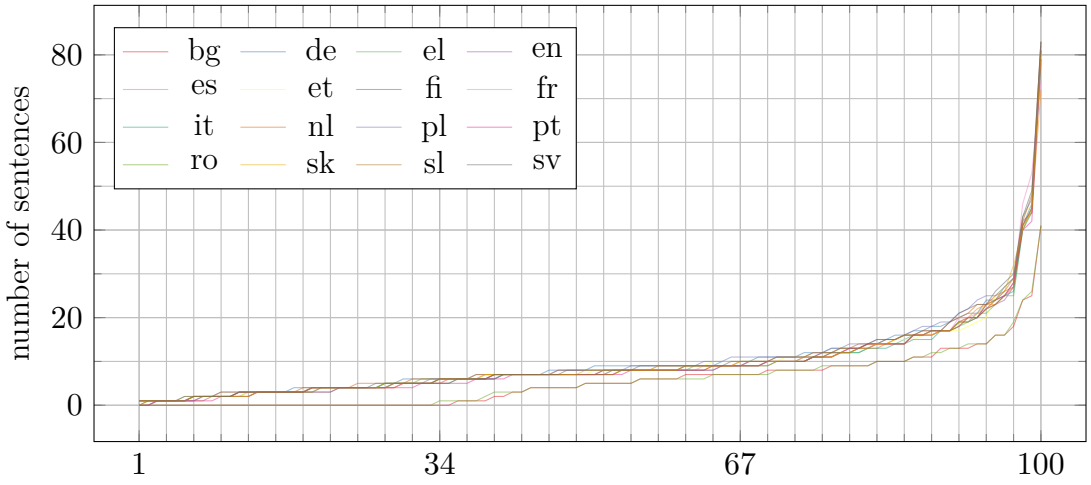


Figure 4.7 – Number of sentences per text in our set of gold alignments.

The 100 sentence-aligned texts yield 1576 AUs: 1369 leaf AUs contain 10.9 (median 14) and 207 branch AUs contain 25.1 (median 24) sentences on average. Table 4.4 shows an excerpt of one of the texts in six instead of the 16 available languages. It contains five leaf AUs (1 to 5) and one branch AU (6), which comprises the AUs 2 to 4. Sentences are typically longer than those shown in this example.

Methods

We perform two evaluations: First, we measure the coverage of generated multilingual AUs in comparison with our gold alignments and, second, we calculate pairwise F-Score measures over all language pairs.[31]

Given a gold AS ($\mathcal{G}$) and a test AS generated by our algorithm ($\mathcal{T}$), the multilingual evaluation compares each AU of the gold AS ($G \in \mathcal{G}$) with the respective best matching AU of the test AS ($T \in \mathcal{T}$). Our measure for the alignment accuracy is the ratio of shared and distinct sentences in both ASs:

$$q(G,T) = \frac{|G \cap T|}{|G \cup T|} \tag{4.16}$$

The best matching AU is the one that gives the highest alignment accuracy value. We calculate the average over all gold AU also taking into account the number of AUs generated:

[31] Since we have no preference for either precision or recall in this case, F-Score hence refers to the balanced F_1-Score.

$$\xi(\mathcal{G},\mathcal{T}) = \frac{\sum\limits_{G_i \in \mathcal{G}} q(G_i, \underset{T_j \in \mathcal{T}}{argmax}\, q(G_i, T_j))}{|\mathcal{G}|} \tag{4.17}$$

This alignment score does not per se differentiate between leaf and branch AU. By using the subset of leaf or branch AUs of both $\mathcal{G}$ and $\mathcal{T}$, we obtain separate scores for leaves and branches. If an algorithm is particularly good at identifying the smaller leaf AUs, but fails at assembling them together to bigger branch AUs, we thus expect $\xi(\mathcal{G}^{leaf}, \mathcal{T}^{leaf})$ to yield a better score than $\xi(\mathcal{G}^{branch}, \mathcal{T}^{branch})$.

With the alignment score ξ as defined above, we are able to compare a generated AS and the gold AS of a particular text. To our knowledge, we are the first to present a hierarchical multilingual alignment approach and the corresponding gold data, which means that this evaluation metric will only become useful for comparison in connection with future approaches.

This hindrance is what we aim to address with our second evaluation. By calculating pairwise F-Scores, we compare multilingual AUs with bilingual ones (i.e., a set of n:m AUs) produced by common sentence alignment tools.[32] These bilingual AUs can be extracted from the multilingual ones by using only those AUs that possess sentences in both languages of the respective language pair and, by subsequently removing duplicates.[33] We refer to them as minimal AS of a multilingual AS for a given set of languages L, which, in theory, can include any number of languages, but, for comparison with bilingual sentence alignment, is limited to two languages: $\mathcal{A}^L$ with $|L| = 2$

For a particular language pair L, we calculate the F-Score of a test AS ($\mathcal{T}^L$) given the gold AS ($\mathcal{G}^L$) for the same text as the ratio of AUs to be found in both sets and the average of both sets' cardinalities:[34]

$$F(\mathcal{G}^L, \mathcal{T}^L) = \frac{2 \cdot |\mathcal{G}^L \cap \mathcal{T}^L|}{|\mathcal{G}^L| + |\mathcal{T}^L|} \tag{4.18}$$

The relation between the definition of the F-Score measure by means of set cardinalities, which corresponds to the Dice similarity coefficient, and the arguably more-widespread definition by means of events (true positives, false positives and false negatives) is depicted in Table 4.9 in Section 4.4.2. The correspondence of

[32] It also helps to identify differences of alignment performance between languages.

[33] Duplicates may result from branch AUs that extend a pairwise leaf AU of the languages in question with leaf AUs only containing sentences in other languages.

[34] Considering matching sets instead of single alignment links, this measure is similar to the translation unit error rate introduced by (Søgaard and Kuhn 2009) with the objective of only rewarding the alignment algorithm if larger structures, translation units, have been identified correctly.

both definitions is also shown in Appendix B.1. True positives are those AUs that are found in both test and gold AS ($\mathcal{G}^L \cap \mathcal{T}^L$), while false positives and false negatives are only found in the test and the gold AS, respectively.

There are two ways to calculate an F-Score measure on a set of texts instead of a single one. We can simply calculate the average of text-wise derived F-Scores, this is the so-called macro F-Score, or we calculate the F-Score on the sum of set cardinalities, this is the so-called micro F-Score. Equations 4.19 and 4.20 show both definitions for a set of gold and test AS pairs ($\mathfrak{S}^L$).

$$F_{macro}(\mathfrak{S}^L) = \frac{\sum_{(\mathcal{G}^L,\mathcal{T}^L)\in\mathfrak{S}^L} F(\mathcal{G}^L,\mathcal{T}^L)}{|\mathfrak{S}^L|} \tag{4.19}$$

$$F_{micro}(\mathfrak{S}^L) = \frac{\sum_{(\mathcal{G}^L,\mathcal{T}^L)\in\mathfrak{S}^L} 2\cdot|\mathcal{G}^L \cap \mathcal{T}^L|}{\sum_{(\mathcal{G}^L,\mathcal{T}^L)\in\mathfrak{S}^L} |\mathcal{G}^L| + |\mathcal{T}^L|} \tag{4.20}$$

The macro average treats all texts the same, no matter how long they are in terms of sentences; the micro average, on the other hand, attaches a substantial greater value to longer texts. The case of a single erroneous alignment boundary, for instance, a 2:1 AU followed by a 1:2 AU that has been identified as a 1:2 AU followed by a 2:1 AU, results in 2 AUs missing in the test AS that are present in the gold AS (i.e., $|\mathcal{G}^L \cap \mathcal{T}^L| = |\mathcal{G}^L| - 2$). For a text with 10 sentences in both languages, this leads to an F-Score of 0.8, while the F-Score of a text with 20 sentences is 0.9. The macro average of both F-Scores, the average of both numbers, is 0.85, while the micro average on both pairs of sets yields 0.87 ($\frac{8+18}{10+20}$). This is due to higher absolute numbers of the larger text, to which more importance is attached by the micro average F-Score measure.

To find out how to set the respective weights for the features of our multilingual sentence alignment algorithm in relation to each other, we use the random walk Metropolis algorithm (Metropolis et al. 1953; Sherlock et al. 2010) in the state space[35] spanned by all 13 feature weights. We limit each dimension to the interval from 0 to 1 and normalize the values such that the length of the resulting vector equals 1, that is, we perform a linear projection of a point in the state space to the hypersphere with radius 1. This is motivated by Equation 4.14, which describes the linear combination of weights to the alignment score that is used for clustering. Multiplying all weights with the same numeric value has no impact on the ranking of association scores.

[35] Also referred to as configuration space by (Metropolis et al. 1953).

Metropolis is a Markov chain Monte Carlo (MCMC) algorithm, which means it randomly draws samples from a probability distribution (Monte Carlo) and each sample only depends on its predecessor (Markov chain). This sample sequence generated by a stochastic process (i.e., the Markov chain) is guaranteed to converge to the probability density π of the state space. Starting with a random position in the state space ($\mathbf{X^0}$), the following states ($\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^t$) are generated as follows:

1. We draw a random variable ϵ from a multivariate distribution – in our case a Gaussian one –, which corresponds to a proposed movement in the state space: $\mathbf{Y}^t := \mathbf{X}^{t-1} + \epsilon$

2. This proposal is, by design of the algorithm, not always accepted.[36] We calculate the acceptance probability α as the ratio of the proposed and the last state's probabilities, limiting it to 1:

$$\alpha(\mathbf{X}, \mathbf{Y}) = min\left(1, \frac{\pi(\mathbf{Y})}{\pi(\mathbf{X})}\right)$$

   If the proposed move represents an improvement or both states evaluate to the same value ($\alpha = 1$), we always accept it. Otherwise, we only accept it if a draw from a single random variable in the interval $[0, 1]$ yields a value lower than α. Accepting means that the new state $\mathbf{X}^t$ becomes the proposal $\mathbf{Y}^t$. In case of rejection, we do not move in the state space and, hence, the new state is the same as the previous one: $\mathbf{X}^t = \mathbf{X}^{t-1}$

The Gaussian distribution around state $\mathbf{X}^t$ proposes closer points in the state space with a higher probability, that is, shorter moves are generated with a higher frequency than longer ones. If a chosen target coordinate in one dimension is outside of the range [0,1], we reflect it back inside this range to ensure that a move in one direction is as probable as a move in the reverse direction. This symmetric proposal distribution is a requirement for the Metropolis algorithm.[37]

The motivation behind not only accepting better (i.e., more probable) states and thus eventually reaching the best one is that there may be multiple maxima. Only accepting better states corresponds to the hill-climbing algorithm, which, depending on the random initial state, cannot reach the global maximum. By accepting worse (i.e., less probable) states with the probability α, we allow the algorithm to explore more regions in the state space without being completely random at selecting states (like a pure Monte Carlo method). The states that we

[36] Otherwise, the sequence of states would correspond to a random walk.

[37] If we had a non-symmetric proposal distribution, that is, a move between two points in one direction is more probable than the other way round, we could use the more general Metropolis-Hastings algorithm (Hastings 1970; D. D. L. Minh and D. L. Minh 2015) instead.

visit over time approximate the weights' joint probability distribution. In addition to that distribution that we construe from a sufficiently large number of samples, we also keep track of the best-scoring position in the state space that we have visited over the course of the process.

Metropolis et al. (1953) use the energy difference in a system of molecules to model the acceptance probability α. The probability of a state is higher for lower energy levels (up to a probability of 1 for zero energy) and lower for higher values. If a proposed state change yields a lower energy level, the ratio $\pi(\mathbf{Y})/\pi(\mathbf{X})$ is greater than 1 and, therefore the change is accepted in any case.

We use the macro average F-Score as quality measure of the respective states. Unlike Metropolis et al.'s energy measure, which has no theoretical upper limit, the F-Score is limited by design to numeric values between 0 and 1. An average F-Score of 0 can only be obtained if every text's F-Score for all language pairs yields 0 and this, in turn, is only possible if the algorithmically generated AS shows no single intersection with the gold AS. In practice, even randomly chosen states frequently yield F-Scores above 0.9.[38] Due to the requirement of monotonicity, the alignment algorithm manages to take the right (clustering) decisions in many cases even if the weighting of the respective features is not optimal.[39]

If we were to use the raw F-Score ratio as acceptance probability, we would accept proposed states with an F-Score that is approximately 1 % lower than the F-Score of the previous state (e.g., a move from 0.95 to 0.94) in about 99 % of the cases ($\alpha \approx 0.99$). This high acceptance rate leads to a slow approximation of the real probability distribution as states with lower values will be accepted frequently and better states thus need more time (i.e., iterations) to be explored. The optimal acceptance rate for the Metropolis algorithm has been shown to be approximately 0.234 (Roberts et al. 1997; Sherlock et al. 2010). We have two options at our disposal to adjust the acceptance rate: On the one hand, we can vary the standard deviation of the multivariate distribution that we draw ϵ from. A larger standard deviation results in larger move proposals and therefore lowers the acceptance rate. On the other hand, a constant $k \gg 1$ exponentiating the F-Scores will move probability mass from the underused low F-Scores to higher ones, such that a small difference in high F-Scores between proposal and previous state entails a larger probability of rejection.

We experimentally found that a standard deviation of 0.4 in combination with the exponent k set to 400 leads to an acceptance rate close to the optimal one. In a scenario where the proposed state's F-Score is approximately 1 % worse than

[38] Note that the same number of correct AUs (true positives) and errors (false positives and false negatives) corresponds to an unbiased F-Score of 0.6667 ($\frac{2}{3}$).

[39] Three texts only comprise a single sentence in all languages; they will form an AS with a single AU, independent of the weights provided.

the previous one (e.g., $\frac{0.94}{0.95} = 0.9895$), the probability that this proposal will be accepted is considerably lower in this setting ($\approx 1.5\,\%$); a reduction of $0.1\,\%$ will be accepted in $66\,\%$ of the cases, a reduction of $0.01\,\%$ in $96\,\%$.

To evaluate our algorithm, we perform a 5-fold cross validation on our gold data. For that purpose, we randomly segment the 100 manually aligned texts into five test sets comprising 20 texts each. Sampling with the Metropolis algorithm as described above is performed on the training sets composed of the respective four remaining parts (80 texts per training set). A pair of training and test set constitutes a fold. We obtain optimal weights per training set by looking up the median value of each weight in a large number of samples observed during the process. Alignment on the test sets is subsequently performed with the optimal weights of the corresponding training set. For comparison, we also align the test sets with hunalign.

Results

We present figures for the aforementioned evaluation in Table 4.8. The first and the third column show F-Scores obtained by applying our algorithm on test and training sets, respectively, using the combination of optimal weights that we gained from 10 000 samples of the generated Markov chains. In the fifth column, we list F-Scores obtained by hunalign on the respective test sets.[40] The intermediate columns show the difference between adjacent F-Scores.

Comparing the performance of our algorithm on training and test sets for both micro and macro average F-Scores, we see that the test sets score slightly lower on average. The absolute difference between average scores on training and test sets, however, is considerably lower than the standard deviation of those differences, which allows for the conclusion that there is no significant performance drop when the weights optimized on the respective training sets are applied to their corresponding test sets.[41] Comparing the performance of our algorithm and hunalign on the test sets, we re-encounter the same situation. Although, on average, our approach yields better micro and macro average F-Scores, the performance on the respective folds is mixed and does not allow to conclude that our approach would yield better pairwise alignments.[42]

A peculiarity is the performance drop of Fold 2 from training to test set, which we observe for both types of F-Scores. While further investigating the composition

[40] We found that hunalign's performance on most language pairs can be improved considerably by providing well-curated dictionaries. Since we do not possess such dictionaries for all language pairs, we let hunalign bootstrap them on the respective parallel texts (see page 70).

[41] Otherwise, we could suspect the optimization process to overfit the training data.

[42] Note that our approach does not produce pairwise alignments in the first place, but multilingual alignments that are projected to minimal pairwise alignments for each language pair.

Table 4.8 – F-Scores attained on the training and test sets of each fold by our multilingual sentence alignment algorithm (mlsa). The performance on the test sets is also compared to hunalign's performance on the same data. The intermediate columns show the difference between adjacent F-Scores. Average ($\overline{F}$) and standard deviation (σ) of the values obtained for all folds are given below.

F_{micro}	mlsa			ΔF	hunalign
	training	ΔF	testing		
Fold 1	0.9515	−0.0079	0.9436	−0.0033	0.9403
Fold 2	0.9551	−0.0251	0.9299	+0.0074	0.9373
Fold 3	0.9469	+0.0061	0.9530	−0.0187	0.9343
Fold 4	0.9453	+0.0038	0.9491	−0.0013	0.9478
Fold 5	0.9467	+0.0055	0.9521	−0.0131	0.9391
$\overline{F}$	0.9491	−0.0035	0.9456	−0.0058	0.9398
σ	0.0041	0.0134	0.0095	0.0102	0.0050

F_{macro}	mlsa			ΔF	hunalign
	training	ΔF	testing		
Fold 1	0.9443	+0.0209	0.9651	−0.0260	0.9392
Fold 2	0.9544	−0.0368	0.9175	+0.0019	0.9194
Fold 3	0.9427	+0.0168	0.9595	−0.0255	0.9340
Fold 4	0.9478	−0.0178	0.9300	+0.0033	0.9333
Fold 5	0.9459	+0.0026	0.9486	−0.0170	0.9316
$\overline{F}$	0.9470	−0.0029	0.9441	−0.0126	0.9315
σ	0.0045	0.0243	0.0200	0.0144	0.0073

of the respective folds, we found that the test sets of this fold comprises the longest text in terms of sentences (the one at x-coordinate 100 in Figure 4.7). That text however, despite being considerably longer than all the others, has median F-Scores in relation to its test set (both scores are approximately 0.94).

Figure 4.8 shows the resulting F-Scores for each fold's test set. The distinguishing property of the text at position 20 in Fold 2 is not its length, but the deviation in lengths: In Portuguese, we only find 3 sentences with 56, 100 and 142 tokens; in Dutch, we find 13 sentences having between 9 and 49 tokens. Our algorithm fails to identify all three gold AUs, which, projected to the language pair Portuguese/Dutch, would be a one-to-two, a one-to-four and a one-to-seven alignment. Since no AU has been identified correctly, the F-Score is accordingly

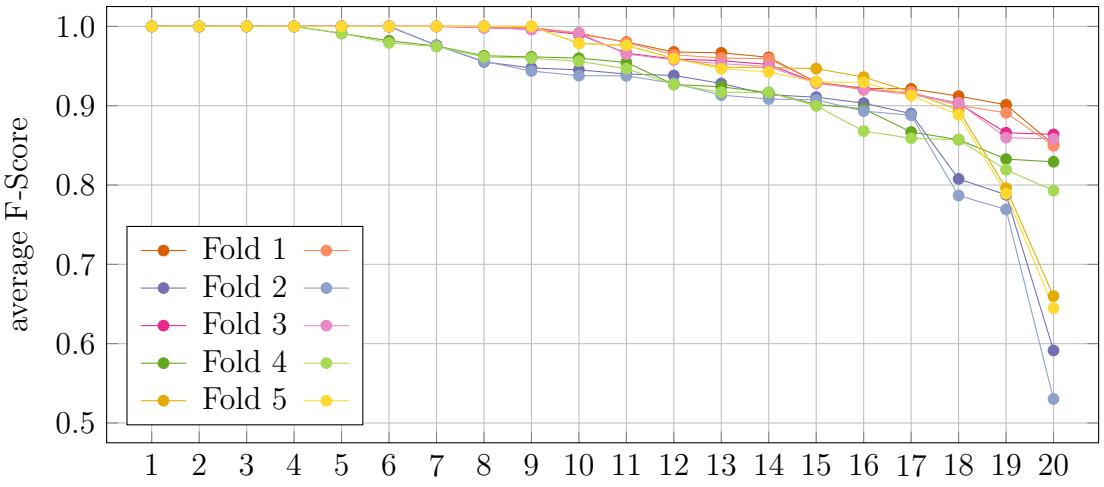


Figure 4.8 – F-Scores per test set in decreasing order. Darker colors denote micro and pale colors macro F-Score averages (average over all language pairs).

0 (for this and nine other language pairs). The worst-scoring text in Fold 5 at position 20 is similarly difficult to align. In that text, we have between one and three sentences per languages, which means that a single erroneous clustering decision makes the difference between an F-Score of 1 and 0. We count 29 times an F-Score of 0, 10 times of 0.6667 and 52 times of 1.

By reason of the small number of texts per test set (20), a single outlier can account for a noticeable difference in the set's performance.[43] In both cases, hunalign performs better, which is presumably due to the influence of erroneous clustering steps on subsequent steps involving other languages.

These observations lead to the question if – apart from being an inevitable task for obtaining multilingual alignments – bilingual alignment also benefits from the inclusion of other languages into the alignment process. To address this question, we compare the F-Scores obtained from applying our algorithm to a particular language pair and applying it to all available languages with a subsequent reduction of the obtained AS to the minimal AS of that language pair. The results are visualized in Figure 4.9.

Except for two language pairs (Greek/Polish for micro average and Greek/English for macro average F-Score differences), direct alignment of two languages performs better than multilingual alignment with subsequent extraction of pairwise minimal alignments. The most prominent is the language pair Estonian/Swedish, where bilingual alignment yields F-Scores that are almost 0.07 higher than the F-Scores

[43] We see similar results for this text in the training sets of the other folds.

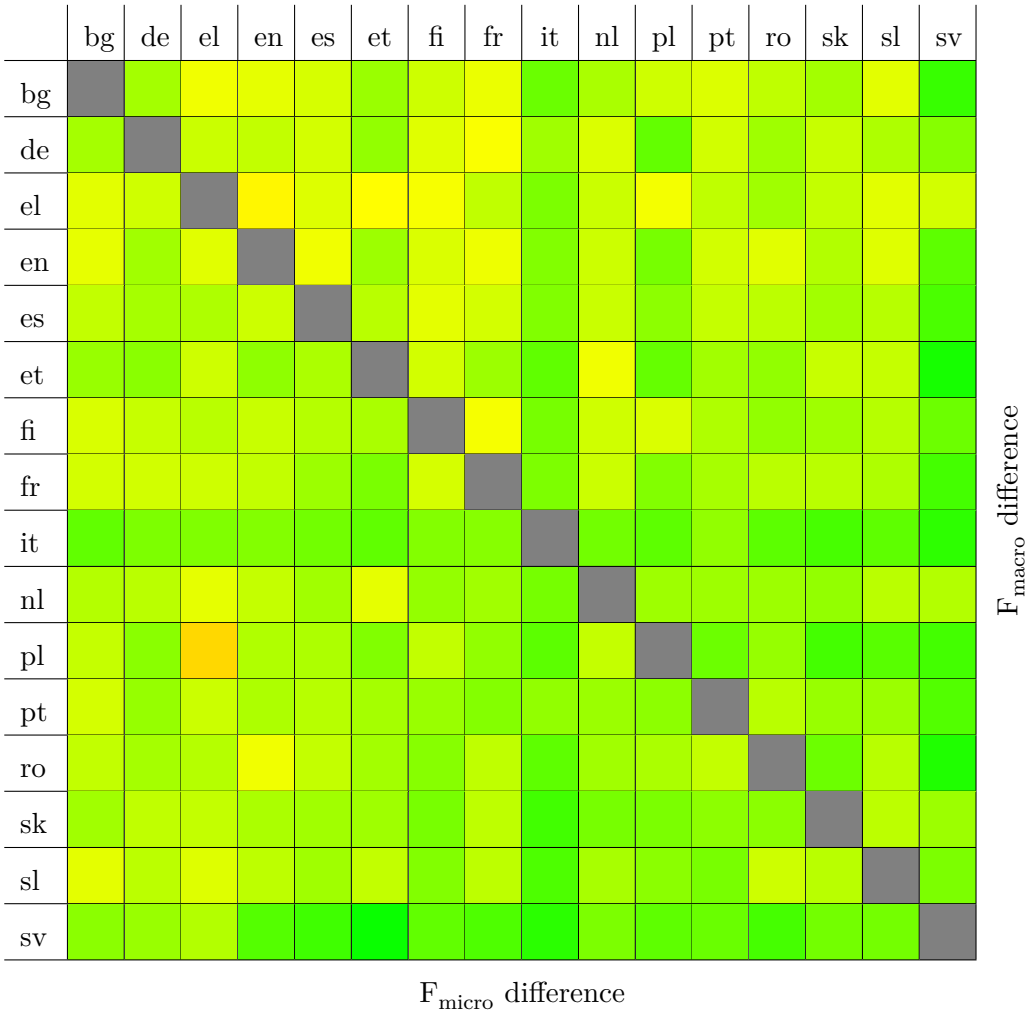


Figure 4.9 – Differences between F-Scores obtained by bilingual vs. multilingual alignment with our multilingual alignment algorithm. Numerical differences are projected to colors from green (bilingual alignment is better) to yellow (neutral) to red (multilingual alignment is better).

obtained from the minimal AS of multilingual alignment.

There are many more clustering decisions involved in multilingual alignment with a higher number of languages and errors can propagate if they happen early in the process, that is, if a link between two sentences is overvalued considerably. In Figure 4.10, we compare the performance of bilingual alignment to trilingual alignment using the micro average F-Score.[44] Unlike the comparison shown in Figure 4.9, we do not aggregate the results to a single average F-Score value, but

[44]Results for the macro average F-Score look similar color-wise.

	bg	de	el	en	es	et	fi	fr	it	nl	pl	pt	ro	sk	sl	sv
bg			ro		el	en		el			en	en		pt	en	es
de	nl		sl	pl	sl		en	pl	el	sl	en	sl	pl	ro	pl	pt
el	es	sv		de	pt	en	sk	ro	sk		en	ro		en	de	en
en	ro	sv	pl		fi		sl	pt	pt	it	fr	sv		et	sv	it
es	pt	et	pl	fr		en	en	ro	en	sl	nl			et	pt	it
et	nl	it	fi	sv	sv		en	pt	pt	en	sv	de	en	en	sv	
fi	sv	sv	nl	nl	pt	el		pt	en	sl	sl	el		ro	fr	en
fr	nl	et	fi	fi	nl	nl	pl		es	en	en	ro	el	es	ro	et
it	nl	fr	pl	sv	el	sl	nl	el			ro	es				en
nl	it	el	fi	fi	fi	it	it	sv	fi		en		bg	sl	sk	sk
pl	et	it		fi	fi	fi	de	fi	fi	fi		sl	nl	sl	pt	nl
pt	et	nl	nl	et	et	nl	de	nl	pl	fi	fi			sv	sv	sl
ro	de	et	pl	fr	de	it	it	nl	nl	de	it	en				it
sk	it	nl	pl	nl	fi	el	nl	fi	pl	de	es	de	nl		et	nl
sl	nl	fr	nl	it	it	nl	de	fi	et	et	it	et	de	pl		en
sv	pl	el	pl	de	fr	fi	pl	de	nl	el	it	nl	fi	el	el	

better F-Scores obtained with bilingual alignment

better F-Scores obtained by means of a third language

Figure 4.10 – Differences between macro average F-Scores obtained by bilingual vs. trilingual alignment. Green stands for better scores obtained from bilingual alignment, red for better scores by trilingual alignment. The third language that accounts for the biggest positive or negative difference is indicated.

examine, which third languages lead to the biggest positive and negative differences with regard to the score obtained by mere bilingual alignment.

We observe that sentence alignment on the language pair Greek/Polish, which even shows a better micro average F-Score in Figure 4.9 for multilingual alignment on all available languages, always improves when any of the other 14 languages is included (this is why there is no language favoring the bilingual alignment in Figure 4.10). Results for other language pairs are ambivalent: They show better scores for one method with one particular language, but also better scores for the

other method with another language. The alignment of German/Greek texts, for instance, can be improved most by including Slovene; Swedish as third language, on the other hand, will impair the performance of our alignment algorithm most.

In general, we see that English is the single most frequent language that improves bilingual alignment of other language pairs. On the other hand, we can conclude that Dutch and Finnish are better left out in most cases if not needed as they will probably lead to a worse alignment. Apart from these observations, the results do not show other noticeable patterns (e.g., regarding language families). To find out, why and how, for instance, Slovak helps to align Dutch and Swedish, we would need to analyze the differences between clustering decisions in corresponding ASs.

The diverging results from Table 4.8 and the missing pattern in Figure 4.10 suggest that there might be insufficient data from which to learn optimal weights. Although the texts from our gold standard differ considerably in length as shown in Figure 4.7, we have seen that this has little influence on the results; larger variation in numbers per AU (e.g., a one-to-seven correspondence), however, renders the alignment process considerably more difficult. To address this question, we look at the data produced by the Metropolis algorithm: Figure 4.11 shows Gaussian kernel density estimates $\tilde{w}_x$ (i.e., an approximation of the probability density estimated from samples) of the respective weights for all folds and the entire data set. We used every 10th sample from a total of 10 000 samples per fold to counteract the autocorrelation of the Markov chain.

Based on the circumstance that most shapes for the same weights (i.e., the same rows) look similar and the obtained optimal weights (i.e., the solid lines) show approximately the same value, we conclude that the 80 texts we use for training on each fold are sufficient to obtain weights that generalize well.[45] Moreover, the investigation of anomalous results in the test set showed that some (few) texts are difficult to align by reason of their inner structure. Even if we include the test set in the training data (i.e., performing the evaluation on weights optimized on all data), improvements on the measured F-Scores are marginal.

The probability distributions per feature weight in Figure 4.11 illustrate the joint probability distribution of all weights. They do not allow for statements regarding the suitability of each respective feature for multilingual sentence alignment, though, as the values are not comparable, that is, a feature that slowly approximates the optimal value of 1 will require a higher weight than another feature that only differentiates between absence (0) and presence (1) if both are supposed to be equally important.

[45] Even though most shapes of the same weight look similar, we see deviations, for instance, in $\tilde{w}_{l1}$ in Fold 5, whose shape, in contrast to other folds' shapes, indicates no clear preference for values below or above the neutral one.

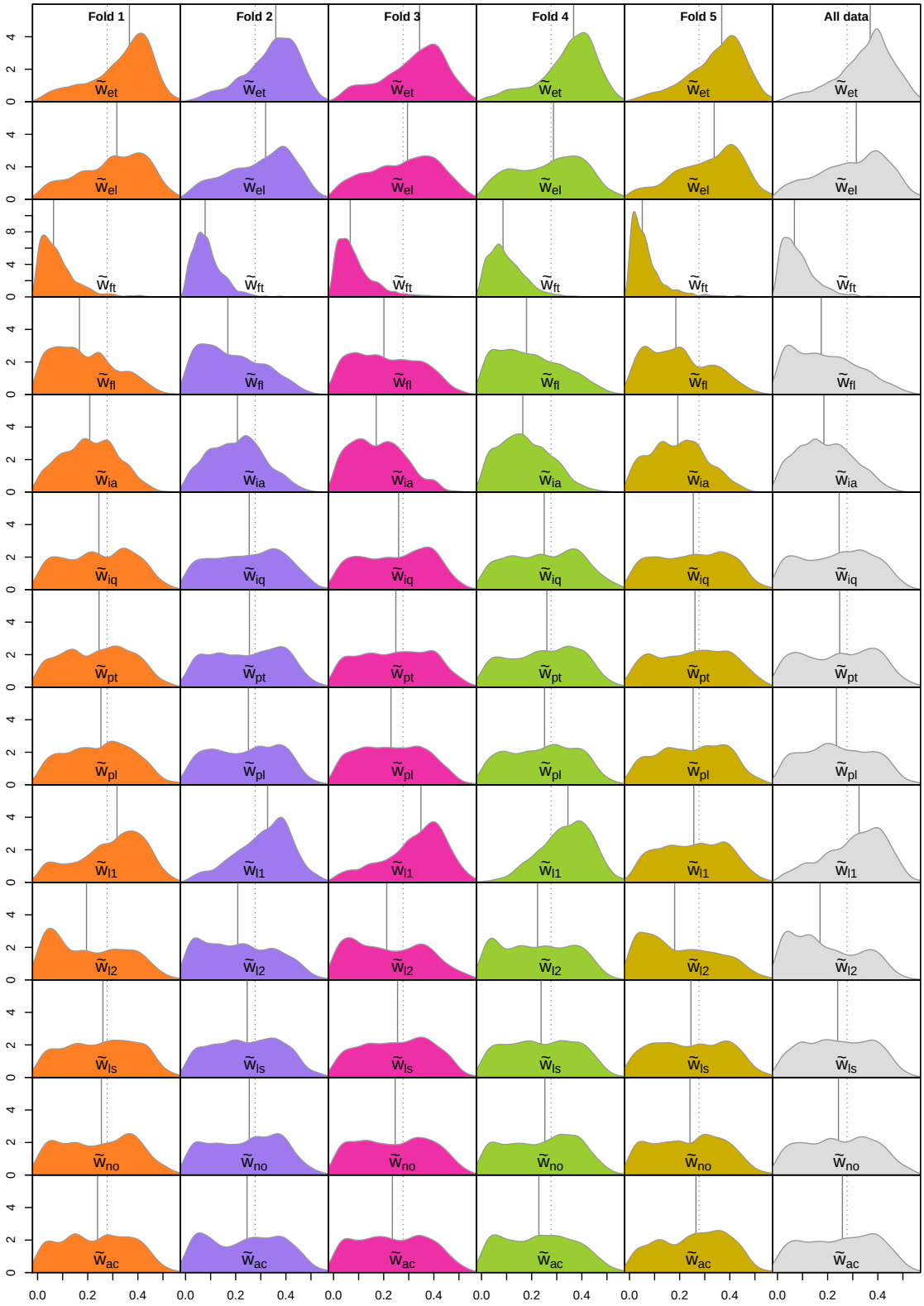


Figure 4.11 – Kernel density estimates of the respective features' distributions for all folds and the whole gold alignment set. The solid lines represent median values; the dotted lines stand for an equal weight distribution.

Nonetheless, if two features use the same underlying formula, we can infer from the probability distributions which one provides more evidence and is thus preferable. The externally generated pairwise alignments on word forms, for instance, is assigned more weight than similar alignments on lemmas, while matching first tokens as approach to capture discourse markers works better with lemmas than word forms of those tokens. We also see that question marks provide more evidence for correspondence than sentence final punctuation in general.

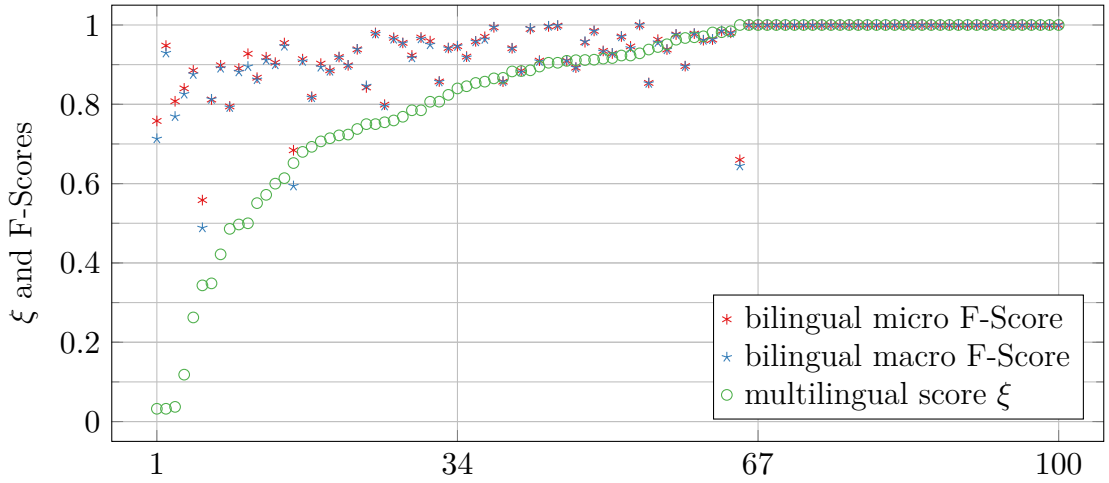


Figure 4.12 – Multilingual alignment scores ξ for all 100 manually aligned texts (green circles). Red asterisks represent the corresponding micro F-Score, blue stars the macro F-Score.

Using the proposed multilingual evaluation function ξ, we obtain scores between 0.0323 and 1 with approximately one third of the texts showing no error (i.e., yielding a score of 1). The median score (0.9159) is considerably higher than the average score (0.8322), which is due to poor performance of the multilingual alignment algorithm on some texts. Figure 4.12 depicts the distribution over all 100 texts.

In general, both F-Score measures and ξ agree on assigning good scores to well aligned texts. It becomes apparent, though, that we measure different aspects of alignment quality with F-Scores compared to ξ when inspecting cases with diverging results. For the text at position 65, for instance, the gold AS consists of a single AU comprising all sentences in all languages, which has been correctly identified by our approach. Minimal AUs, however, are incorrect for many language pairs in this case. Conversely, the text at position 2 shows good overall performance on the identification of pairwise AUs, but disagrees with the gold standard regarding the composition of multilingual AUs.

4.4 Word Alignment

Aligned sentences provide the basis for word alignment. In theory, word alignment algorithms would also work on parallel texts. However, we expect corresponding words to be found in corresponding sentences, or, the other way round, sentence alignment is designed to identify sentence correspondences such that these alignments enclose corresponding words.[46]

Limiting word alignment to previously aligned sentences implies two essential advantages from a computational point of view: On the one hand, alignment errors due to the aligned words not being contained by corresponding sentences are rendered impossible; on the other hand, we reduce computational complexity, which, when assessing all possible alignments, increases quadratically with the input length in both languages.[47]

Free word order is the most important factor in making word alignment a more challenging task than sentence alignment. We have seen that sentence alignment algorithms assume monotonicity (see Section 4.2.1), that is, that the meaning in parallel texts is conveyed in the same order in all languages. At least in the case of the European Parliament debates, translators typically only apply small changes to texts with regard to the sub-division into sentences, for example by joining or splitting them (e.g., by means of a conjunction or a period). Larger deviation between texts in two languages (e.g., German and Spanish shown in Table 4.2) typically stems from independent translations from a third source language.

The Rhetorical Structure Theory (RST) (Mann and Thompson 1987, 1988) describes the structure of texts on the level of discourse. Its units are sentences and subclauses of sentences. Figure 4.13 shows a parallel sentence pair where the corresponding clauses are color-coded. An RST structure is a hierarchical tree, which typically comprehends the whole discourse of a text. The RST analysis of both sentences are similar in terms of relations though they differ in the order of their clauses. The example reveals that we cannot assume monotonicity for the subclauses of a sentence.

Alongside variable structure of sub-clauses within sentences, part-of-speech and syntactical choices vary considerably between languages (see example in Figure 4.14). Moreover, lexical choices are not guaranteed to be consistent between translations. To demonstrate lexical deviation of translators, we queried the FEP6 corpus in Multilingwis (see Section 5.2) for the English term ‘key point’ and Eng-

[46] Or as Kay and Röscheisen (1993) put it: “a pair of sentences containing an aligned pair of words must themselves be aligned.“

[47] For multilingual alignment, the complexity would grow polynomial with the number of languages as exponent.

English Indeed, it is by taking real action closer to citizens, by simply talking to them about Europe, that they can get a clearer picture of what the European Union does for them in their daily lives.

German Tatsächlich kann man den Bürgern nur ein klareres Bild davon vermitteln, was die Europäische Union für sie in ihrem jeweiligen Alltag tut, indem man ihnen mit echten Aktionen näherkommt und einfach mit ihnen über Europa spricht.

Figure 4.13 – Corresponding parts of a parallel sentence. The clauses highlighted in orange and yellow explain by means of what the citizens ('they') would 'get a clearer picture'. Following the rhetorical structure theory (RST), one would thus connect both clauses (satellites) to the green clause (nucleus) using the 'Means' relation. The green clause, in turn, connects to the previous sentence using an 'Elaboration' relation as it elaborates on what has been said before.

lish as source language.[48] As result, we get six frequent (i.e., more than three occurrences) translation variants in Finnish ('avainkohta', 'tärkeä kohta', 'keskeinen seikka', 'keskeinen kohta', 'keskeinen asia' and 'avainasia'), five in German ('wichtiger Punkt', 'Kernpunkt', 'zentraler Punkt', 'Hauptpunkt', 'wesentlicher Punkt'), four in French ('point clé', 'point essentiel', 'points-clés' and 'point important') and Italian ('punto chiave', 'punto principale', 'punto fondamentale', 'punto essenziale') and three in Spanish ('punto clave', 'punto fundamental', 'aspecto clave'). In total, we see 41 search hits.

Literal translations of 'key points' into the three Romance languages are 'point clé' (11 occurrence), 'punto chiave' (16 occurrences) and 'punto clave' (17 occurrences). These are also the most frequent translation variants per respective language.[49] The number of search hits where all three translations appear together, however, is smaller than expected; we can expect at most 11 cases, but there is only one (9 %). For the language pair Italian/Spanish, we see 4 out of 16 (25 %), for French/Spanish 4 out of 11 (36 %) and for French/Italian 5 out of 11 (45 %). The distribution of translation variants indicates that lexical choice – provided that there are semantically close alternatives such as in the case of 'key points' – depends considerably on the respective translator's preference.[50]

[48] The translated sentences are direct translations since English is one of the European Parliament's relay languages.

[49] When we do not restrict the source language to English, the most frequent translation variant for French is 'point essentiel' with twice as many occurences as 'point clé'.

[50] Brown, V. J. Della Pietra et al. (1993) comment on that topic: "A string of English words, e, can be translated into a string of French words in many different ways. Often, knowing the broader context in which e occurs may serve to winnow the field of acceptable French translations,

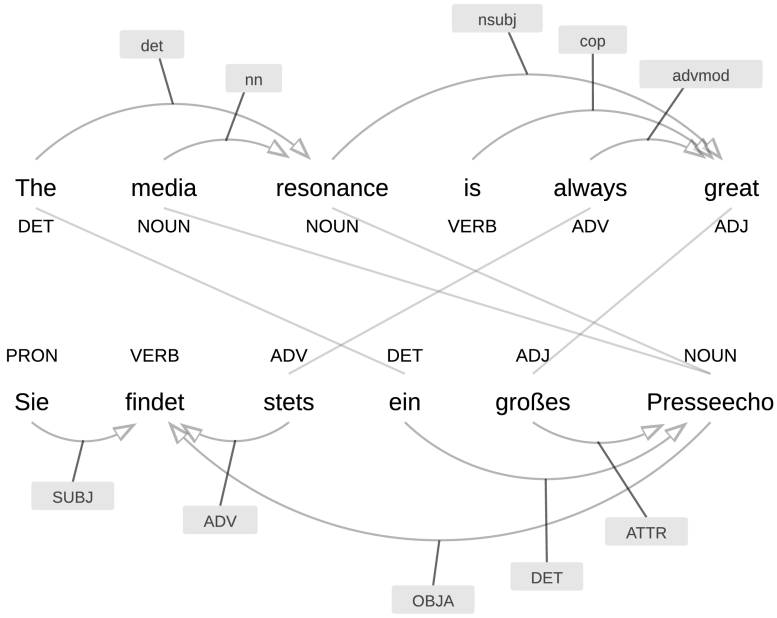


Figure 4.14 – An English/German parallel sentence pair showing syntactic variation. Corresponding tokens are connected by straight lines.

Besides obtaining statistical models of word correspondences for machine translation, a variety of applications for word alignment has been mentioned in the literature. Closely related to machine translation are machine-aided translation and terminology extraction. Bilingual (and multilingual) lexicography are corpus-based fields that rely on corpus statistics and examples, especially 'good dictionary examples' (GDEX) (Kilgarriff, Husák et al. 2008). Good corpus examples also matter for language learners (Volodina et al. 2012). We have compared different parallel corpus query systems also in consideration of their usefulness for language learners (Volk, Graën and Callegaro 2014). Other applications include word sense disambiguation (Diab and Resnik 2002), syntactic transfer (Bouma et al. 2008) and typology studies (Mayer and Cysouw 2012).

4.4.1 Approaches

The word alignment tasks deals with the identification of corresponding tokens in, typically, two parallel sentences. The idea that lead to the introduction of the so-called *IBM translation models*, or IBM models for short, (Brown, V. J. Della Pietra et al. 1993) arose from the availability of parallel corpora and sentence alignment

but even so, many acceptable translations will remain; the choice among them is largely a matter of taste."

to be applied thereupon. These methods provided the source material, parallel sentences, from which to learn word translation probabilities. Word alignment algorithms are designed to search for the most probable configuration of word alignments by maximizing the conjoint probabilities. Later works took on the original concepts and replaced components with more sophisticated ones. Several publications suggest generally looking for larger units and align sequences of words, referred to as phrases, instead of defaulting to single word alignments. Some of these approaches resort to word alignment to base their models thereon.

With recent developments in computer hardware, traditional approaches in statistical machine translation were, to a large extent, abandoned in favor of deep neural network approaches, known as deep learning (see, for instance, Bahdanau et al. 2015). Like traditional machine learning methods, those networks learn from a large amount of human translated sentences. The difference is that they learn in an independent and distributed way, that is, manually engineered features such as the ones learned by the IBM models are not required anymore. If a deep neural network succeeds in 'translating' sentences correctly, it has learned a myriad of miniature features, represented by the neurons, which are distributed over multiple layers and together reproduce the implicit knowledge of the correct translations seen during the training phase.

As beneficial as this architecture is from a machine learning point of view, the drawback is that we cannot educe word alignments from neural machine translation models as they are not explicitly represented. Recent development in seminal word alignment approaches (Gal and Blunsom 2013; Östling and Tiedemann 2016) indicates that there is a continuing interest in word alignment, although the most prominent 'client', statistical machine translation, has dropped out of the game even before deep learning methods became popular.[51]

The IBM Translation Models

The aforementioned IBM translation models consist of five definitions of probability distributions that intent to approximately capture the conditional probabilities of words in two parallel sentences. They have been developed with computational complexity in mind, and thus make some concessions as to linguistic reality, on the statistical model that means to assume independence where dependence is hard to disclaim.

The first model is inexpensive to calculate, but making simplifying assumptions that are known to not hold in the real world, namely that the number of words of both sentences would be independent as would be the position of corresponding words. To this end, model 1 employs uniform probability distributions for both

[51] According to Koehn (2010), "IBM models are no longer the state of the art in translation modeling."

variables. Model 2 also takes into account the absolute positions of each pair of words. It uses the probability distribution calculated by its predecessor. While model 1 only performs translation on a lexical level, model 2 rewards the words that are in the correct positions. For both these models, the authors acknowledge that they "lead to unsatisfactory alignments". They describe them as "spiritually deficient".

The novelty that model 3 introduces is an additional distribution that models the nature of word-translation relationships called *fertility*. The fertility of a word in the source language stands for the number of target language words it translates to. To produce the English expression 'I believe' from Spanish 'creo', we need to generate two target words given 'creo' as source. In cases with more lexical variation (e.g., German 'entsprechend' *'corresponding'* to French 'conforme', 'correspondant' (one word), 'd'autant', 'en conséquence' (two words), ''en fonction de ', ''à la hauteur' (three words), up to 'à la juste mesure de') the fertility distribution reflects lexical preferences of the source word in matters of target word numbers.

Model 4 respects the observation that phrases typically translate to phrases. In the majority of phrases, their words occupy adjacent positions in a sentence. Exceptions are, for instance, the French negation 'ne … pas' where the negation parts enclose the finite verb or fixed expressions with variable parts (e.g., cardinal numbers). Internal reordering of phrases as required, for instance, by noun phrases with adjectives when translating between Germanic and Romance languages is achieved with the help of word classes. Brown, V. J. Della Pietra et al. (1993) define 50 classes as described in (Brown, Desouza et al. 1992). Both adjacent and distant positions of phrase elements are encoded in model 4.[52]

The final IBM model, model 5, straightens up the *deficiencies* of the previous models. Brown, V. J. Della Pietra et al. (1993) define deficiency like this: "When a model has this property of not concentrating all of its probability on events of interest, we say that it is deficient." They assert that "In Model 4, not only can several words lie on top of one another, but words can be placed before the first position or beyond the last position in the French string." Previous models have not been designed to generate usable alignments, but "Models 1-4 serve as stepping stones to the training of Model 5." It is thus IBM model 5 that effectively generates word alignments for us. A more detailed discussion of the IBM translation models can be found in (Koehn 2010, Chapter 4) or (Tiedemann 2011, Section 5.1).

[52] As far as our experience goes, this method does not handle well long-distant cases such as particle verb prefixes in German (see Section 3.2.2).

Extensions to the IBM Models

Most subsequent works on word alignment base on the IBM models and try to improve them by modifying components or adding a layer. Vogel et al.'s (1996) contribution is the use of a Hidden Markov Model (HMM) to model word positions. They argue that "words are not distributed arbitrarily over the sentence positions, but tend to form clusters". The HMM is thus meant to capture "the strong dependence of a_j [the alignment at position j] on the previous alignment". As their approach deals with positioning words in the target language, it can be used to replace the IBM model 2. In (Och and Ney 2000), the standard IBM models and Vogel et al.'s (1996) HMM solution for the word position problem are compared and extended by a dictionary that increments the weight of its entries with regard to the alignment probabilities learned from training corpus.

The *GIZA++* software implementation of the IBM models plus the HMM extension (Och and Ney 2003) established a de facto standard for word alignment and is still widely used.[53] Apart from implementing the existing models, they complement the application with a new model, a combination of HMM and IBM model 4, which they refer to as IBM model 6. By combining both models that target the position of words, in the source and target language, they aim at obtaining better alignments.

GIZA++ makes use of the *expectation maximization (EM)* algorithm (Dempster et al. 1977), which iteratively maximizes the likelihood of the training data, to obtain the optimal alignment also referred to as Viterbi alignment (Brown, V. J. Della Pietra et al. 1993). The algorithm consists of two alternating steps, each of which handles one direction of the dependency between parameters and latent variables of the model.[54] The EM algorithm may, in theory, require an infinite number of iterations to converge. In practice, few iterations are typically sufficient to obtain satisfactory values.

The aforementioned models have one characteristic in common: They generate alignments only in one direction, from source to target language, that is, it is possible that a word in the source language is aligned to multiple words in the target language but not the other way round; each target language word is aligned to exactly one source language word, including the nonexistent 'empty word' that is introduced into every source language sentence exactly to account for the target language words that do not have a counterpart in the source language.

[53] GIZA++ builds on the application Giza by (Al-Onaizan et al. 1999). Development on GIZA++ and its multi-processor variant MGIZA++ seems to have ceased in the meantime, though.

[54] The EM algorithm is not guaranteed to find the global maximum and from model 2 onward we cannot be sure that there is no more than one maximum (Vogel et al. 1996), which means that the result of the EM algorithm can be different for different starting points.

It is easy to show that one-to-many alignments cannot cope with real world correspondences between any two languages. The Spanish expression 'cada vez más' *'increasingly'* can be expressed by 'sempre più' in Italian. It is perfectly acceptable to align 'cada vez' *'each time'* with 'sempre' *'always'* and 'más' *'more'* with 'più' *'more'*. It we compare these expressions with 'de plus en plus' in French, we can align both 'plus' *'more'* with 'más' or 'più', but 'de' *'from'* + 'en' *'to'* is not a match for 'each time' or 'always'. Consequently, we would prefer to align the whole expressions, which gives us a two-to-three alignment (Italian/Spanish), a two-to-four alignment (Italian/French) and a three-to-four alignment (Spanish/French).[55]

A common way to obtain many-to-many alignments from GIZA++ is to train the alignment model in both directions (i.e., to interchange source and target language) and symmetrize the results. *Symmetrization* means to derive symmetric alignments from both lists of asymmetric relations. Different symmetrization techniques have been found to serve for different purposes (for an overview see Och and Ney 2003, pp. 32–33; Koehn, Axelrod et al. 2005; Koehn 2010, Section 4.5.3; Tiedemann 2011, pp. 75–77; Östling 2015, Section 2.3.8.4).

It is plausible that the symmetrization step after alignment adds another potential source of error as Liang et al. (2006) say. This shortcoming of the prevalent IBM models prompted them to develop an alignment model that yield symmetric alignments directly. To this end, they additionally train two HMM models, one for each direction, jointly. That means that during training, their parameter optimization algorithm takes into account the probabilities of all alignments in a sentence suggested by one of the directional models could have been produced by both models.

Unlike the downstream combination of two models by means of symmetrization, their single model generates alignments with an inherent high degree of agreement.[56] According to their evaluation, "intersecting the predictions of two directional models outperforms each model alone." They compared the results of IBM models 1 and 2 and Vogel et al.'s (1996) HMM model once in their hitherto existing variant and once with jointly training of the respective model. That means they integrated their joint alignment model into each one of those models.

As final alignment set (AS) of a sentence, previous approaches resort to its most probable AS, called Viterbi alignment. Liang et al. (ibid.) introduce a variant to construct the resulting AS from comparably good alignment units (AUs), called posterior decoding, which uses a threshold for the posterior probabilities of each

[55] Even if we approved the alignment 'cada vez' and 'de' + 'en', this would be beyond the scope of the IBM models.

[56] We noticed that their application, the Berkeley Aligner, shows a tendency to rather let some words unaligned than to forcefully find an alignment. This is opposite to our observation with GIZA++, where infrequent types tend to act as 'garbage collectors' (Moore 2004).

AU. This variant allows the exclusion of less probable AUs and thus generally leads to a better overall alignment (AS). They implemented both the joint HMM training and posterior decoding in an application called *Berkeley Aligner*.

Alignment of Phrases

A different idea to overcome the limitation to one-to-many alignments of target language words is to align phrases instead.[57] The main advantage is that the alignment problem is reduced to identifying one-to-one alignments of phrases, which means that fertility is not a problem anymore and, in addition, there are fewer items to align (given that words and phrases do not coincide). This poses a new challenge: how to partition sentences into phrases in the first place. Marcu and Wong (2002) present a phrase-based alignment model, which is learned from parallel sentences assuming 'concepts' that are represented in both languages. When applying the model after training, in the decoding step, they chose an initial setup of phrases and alignment and then repeatedly sample by applying small changes to the current configuration until they reach a (local) probability maximum.

DeNero, Gillick et al. (2006) foreshadow the idea of hierarchical alignments when they argue that word and phrase alignments differ from a probabilistic point of view. The former improve with subsequent iterations of the learning algorithm that estimates alignment probabilities while the latter worsen. This is due to the fact that phrase boundaries are also estimated and, with several iterations, converge to a particular optimal segmentation of sentences into phrases. They argue that "if one segmentation subsumes another, they are not necessarily incompatible: both may be equally valid." Approaches that learn segmentation jointly with alignment probabilities (Marcu and Wong 2002, such as) thus tend to overfitting on the training data.[58] Phrase alignment approaches that build on top of word alignments rather than learning phrases do not suffer from this degradation.

DeNero and D. Klein (2008) show that finding an optimal phrase alignment is a complex problem (NP-hard) when all possible ASs of a parallel sentence pair are considered, that is the Cartesian product of all possible segmentations in source and target language that yield the same number of phrases. However, they show how to express the optimization as a well-understood constraint problem (ILP) that can be solved efficiently.

[57] The term phrase is to be understood as any sequence of words as those statistical models have no notion of linguistic content. In fact, Koehn, Och et al. (2003) state: "Learning only syntactically motivated phrases degrades the performance of our systems." The definition of phrases as sequences excludes discontinuous expressions.

[58] Riley and Gildea (2012) characterize the problem of previous approaches to favor longer over shorter phrases as that the longer phrases "explain the training data well but are unlikely to generalize".

Bayesian Models

In (DeNero, Bouchard-Côté et al. 2008), the authors approach the search problem for phrase alignments by letting a *Gibbs sampler*, a Markov chain Monte Carlo (MCMC) method, (for a comprehensive description, see Neal 1993, Section 4) approximate the joint distribution of multiple random variables for estimating phrase pair counts. They start with a random alignment of phrases (i.e., sequences of tokens) as initial state and continue sampling new states by applying small changes to the respective previous state (e.g., exchanging phrases between two AUs, joining or splitting AUs). The advantage of this sampling method is that it is computationally tractable and, unlike DeNero and D. Klein's (2008) approach, guaranteed to converge to the posterior distribution of the model.

Gibbs sampling starts, like the EM algorithm, with an initial configuration with no special requirements apart from being a valid one, that is, both sentences have been segmented such that each word forms part of exactly one phrase. In DeNero, Bouchard-Côté et al.'s (2008) case this means a segmentation of both sentences into phrases and an AS build thereon. The sampling then consists in "repeatedly replacing each component with a value picked from its distribution conditional on the current values of all other components." (ibid.). That is, at each iteration, the configuration is frozen apart from one variable. This variable receives a new value which is exclusively determined by the values of the other variables. After many iterations (an infinite number in theory), they obtain each variable's distribution from the samples collected since "the variable assignments sampled during all iterations will approach the true distribution according to the model" (Östling 2015).

In (Riley and Gildea 2010, 2012), the authors describe their experiments with *variational Bayes*, which is, like Gibbs sampling, a technique to approximate Bayesian inference. For that purpose, they reimplement the maximization step of the EM algorithm in GIZA++ by modifying the formulae that are used to calculate translation (i.e., two words 'translate' to each other) and alignment probabilities (i.e., words on two particular positions are aligned with each other), which are used by all models.[59] The main objective of adopting variational Bayes is to control the effect of overfitting, which becomes manifest in the case of low frequent words being aligned to several words in the other language due to increased likelihood.[60] In conformance with (Mermer and Saraçlar 2011), they infer from their evaluations that Bayesian inference methods outperform the classical EM algorithm.

[59] They follow the HMM implementation with variational Bayes by (Beal 2003, Section 3.4).

[60] Brown, S. A. Della Pietra et al. (1993) themselves state: "Rare words have a tendency to act as garbage collectors in our system."

Another variant of GIZA++ that rests upon Bayesian has been developed by (Gal and Blunsom 2013).[61] They introduce the *hierarchical Pitman-Yor process*, which is a generalization of a Dirichlet process (see Östling 2015, Sections 2.4.4 and 2.4.5) to word alignment, as a replacement for categorical distributions, such as the positions of aligned words. The idea behind Pitman-Yor processes is that they can produce power-law distributions, which are a characteristic of natural language (Goldwater et al. 2006, Chapter 2) and allow to model the probabilities of word sequences with different lengths jointly.

Östling (2015) gives a comprehensive overview of different Bayesian models for word alignment including all the aforementioned concepts. He focuses on the application of Gibbs sampling (see also Goldwater 2007), in particular collapsed Gibbs sampling where some variables are integrated out, which renders the sampling process more efficient. He also experiments – also by means of Gibbs sampling – with multilingual alignment, a challenge that, to our knowledge, only Lardilleux and Lepage (2009) and (Mayer and Cysouw 2012) have met before. The algorithm described in (Östling 2014) for "simultaneous word alignment in massively parallel corpora" introduces statistically generated 'concepts' with which words in the respective languages are aligned.

The word aligner *efmaral* (Östling and Tiedemann 2016) uses a collapsed Gibbs sampling algorithm based on Vogel et al.'s (1996) HMM model plus a model for fertility. Its authors not only report an alignment quality similar to GIZA++ (and according to our evaluation consequently better than fast_align; see below) but also a performance gain as sampling calculations are less complex than previous approaches.[62]

Sub-sentential Alignment

The multilingual word aligner *Anymalign* (Lardilleux and Lepage 2009; Lardilleux, Lepage and Yvon 2011; Lardilleux, Yvon et al. 2012) takes an orthogonal approach to identify corresponding words and phrases. Instead of primarily looking for high frequent word correspondences as stable hubs, it focuses on hapax legomena, those words that only appear once in a text. The idea is presented in (Lardilleux and Lepage 2007).

The authors describe a vector space that is spanned by the parallel sentences as dimensions. Words are expressed by vectors with a positive value for each dimension (i.e., sentence) they appear in; the value represents the number of occurrences in that sentence. The angle between vectors of different languages connotes how well occurrences of the respective words correspond to each other; they call

[61] Unfortunately, the application they describe has never been released.

[62] We have also seen results comparable to GIZA++ in our experiments and efmaral is at least an order of magnitude faster than GIZA++.

this measure translation distance. A translation distance of 0 identifies words that share the same distribution in the parallel sentences provided. Having found alignments in the whole corpus by that means, different subsets of it, called subcorpora, are treated the same way. This method repeats with other randomly sampled subcorpora until terminated by the user. The resulting alignments of words and phrases are accompanied by translation probabilities and lexical weights, both concepts introduced by (Koehn, Och et al. 2003), to measure alignment quality. This approach successfully exploits the Zipfian distribution of words in a corpus, very much like the characteristics of a Pitman-Yor process (Goldwater et al. 2006; Teh 2006).

A completely different case of sub-sentential alignment are *parallel treebanks*. Parallel treebanks have been employed to study syntactic correspondence of two languages. The linking of corresponding leaves (words) and nodes (syntactic constituents) has typically been done manually, for instance with the help of tools such as the *Stockholm TreeAligner*, TreeAligner for short (Volk, Lundborg et al. 2007; Lundborg et al. 2007). TreeAligner allows a human annotator to link words or syntactic constituents of two languages, classifying them into two categories (referred to as 'exact' or 'good' and 'approximate' or 'fuzzy'). It requires parallel sentences with syntactic constituency trees for both languages. The application can also be used to query its own treebanks. Another, more recent tool for queries on aligned parallel corpora is ANNIS3 (Krause and Zeldes 2014), which can also handle treebanks such as the ones produced by the TreeAligner.

The manual creation of treebanks is a time-consuming work. Zhechev and Way (2008) and Zhechev (2009) present an approach, referred to as sub-tree aligner, to automatically generate treebanks from parallel corpora using constituency parsers for both languages and a word aligner.[63] Another approach to build treebanks automatically is described in (Tiedemann and Kotzé 2009a,b). The authors employ a probabilistic model that bases on association features such as the lexical probabilities already used by the sub-tree aligner and combined structural relations from the syntax tree. Unlike the sub-tree aligner, their model needs hand-crafted syntactic alignment to learn from. The application Lingua-Align (Tiedemann 2010) implements their approach.

Our definition of multilingual hierarchical word alignment and the hierarchical structure of treebanks have in common that both build on standard word alignment units that, in turn, form higher level aligned units. In contrast to treebanks, hierarchical word alignment units are not targeted on representing syntactic structures.

[63] They also explain how their method could be adapted to cases where only one language is parsed, which would make their sub-tree aligner a syntactic transfer algorithm.

Other Approaches

Triangulation is using a third language to improve some technique originally applied to two languages. Alongside many other applications, triangulation has been successfully applied to information retrieval (L. A. Ballesteros 2002), annotation transfer of grammatical structures (Bouma et al. 2008), the creation of dictionaries for sentiment analysis (J. Steinberger et al. 2012) and various aspects of machine translation (Cohn and Lapata 2007; Y. Chen et al. 2008; Wu and Wang 2009; Crego et al. 2010). The first reported application of triangulation to word alignment is called *pivot alignment* (Borin 2000a,b) and consists in cascading two previously obtained word AS.[64] Pairwise word alignment is performed on all combinations of three parallel sentences (in the source, target and pivot language). The resulting AS between source and target language is then defined as the union of the AS from the source-to-target language alignment and the AS obtained by intersecting both ASs of source-to-pivot and pivot-to-target language. Since the underlying word aligner segments the parallel sentences into so-called multiword 'link units' (Tiedemann 2000), the pivot alignment method is capable of identifying phrases that have not been found by the direct alignment of source and target language.[65]

A new word alignment method based on the integration of so-called *alignment clues* (also referred to as association clues) is presented in (Tiedemann 2003a, 2004). Clues are indications from different sources of which pairs of words should be aligned with each other. Tiedemann distinguishes declarative clues, which are binary alignment indications coming from linguistic resources, and estimated clues, which, in contrast, are indications from relative measures such as word alignment models. Different clues are combined using source-specific weights. The resulting evidence for each combination of source and target language word can then be used to construct the optimal AS for a particular requirement (in terms of precision vs. recall and focus on single words or multiword units).

Another, more recent, application derived from the IBM models is available under the name *fast_align* (Dyer et al. 2013). The authors argue that models 1 and 2, which rely on sequences and can thus be calculated easily in comparison to the higher-level models, are suboptimal by design. They propose a stand-alone replacement for model 2 with "[e]fficient inference, likelihood evaluation, and parameter estimation algorithms". Although being consistently fast in our experiments (sometimes beaten by efmaral), the alignments calculated by fast_align always turn out to be worst in our evaluation (see below).

[64] The initial word alignment was obtained by means of the Uppsala Word Aligner (Tiedemann 2000; Hein 2002, Section 6.1), which forms part of the Uplug tool (Tiedemann 2002).

[65] As phrases are determined by segmentation, that is, modifying tokenization in hindsight, "only contiguous phrases are identified."

The problem of "translation units that are smaller than a word or whose endpoints are not marked by word boundaries" is treated in (Kay 2004). His work is motivated by the fact that word boundaries are to some extent language-specific. Different languages organize meaning in a different number of words. The noun phrase 'mine-clearing works' corresponds to one word in German ('Minenbeseitigungsarbeiten') and five words in Spanish ('trabajos de la limpieza de minas'), which inspired us to propose a hierarchy to represent multilingual alignments (Graën and Clematide 2015). Kay's (2004) idea is to disregard word boundaries and instead of that first identify substrings to be aligned, for which he avails himself of suffix trees.

4.4.2 Evaluating Word Alignment

Evaluation of word alignment can be divided into those methods that measure the indirect effect of an alignment approach with regard to a particular application or task and those that try to capture alignment quality directly by comparison with an alignment gold standard.[66]

Table 4.9 – Contingency table for alignment evaluation. A set of gold AUs ($\mathcal{G}$) is compared to a set of test AUs ($\mathcal{T}$). If an AU A forms part of both sets, it counts as true positive (TP). True negatives (TN) cannot be expressed in terms of these two sets.

	$A \in \mathcal{G}$	$A \notin \mathcal{G}$
$A \in \mathcal{T}$	TP	FP
$A \notin \mathcal{T}$	FN	TN

(a) Test results as events

	$A \in \mathcal{G}$	$A \notin \mathcal{G}$
$A \in \mathcal{T}$	$\mathcal{G} \cap \mathcal{T}$	$\mathcal{T} \setminus \mathcal{G}$
$A \notin \mathcal{T}$	$\mathcal{G} \setminus \mathcal{T}$	

(b) … and in set notation

For the latter, the most natural way to determine alignment quality is to count an alignment unit (AU) as correct if it appears in both the test alignment set (test AS: $\mathcal{T}$) and the gold alignment set (gold AS: $\mathcal{G}$). Besides those correctly identified AUs, the *true positives (TP)*, we also find AUs that are only present in the gold or the test AS – assumed that the test AS is not entirely correct, which is tantamount to both ASs being equal. These non-matching AUs are named *false positives (FP)*, if they are found in the test AS but not in the gold AS (i.e., the alignment algorithm incorrectly identified an AU), and *false negatives (FN)*, in the opposite case (i.e., the algorithm fails to identify a correct AU). These cases are depicted in Figure 4.9a.

[66] Lardilleux, Gosme et al.'s (2010) proposal of an evaluation method based on comparison with automatically generated bilingual lexicons does not fit well into this scheme. They target improbable language pairs where the creation of gold standard would be exceedingly expensive.

English	Can we afford to risk that kind of relationship?
German	Können wir es uns erlauben, diese Beziehung zu gefährden?
Swedish	Har vi råd att riskera den förbindelsen?

	English	German	Swedish
$\mathcal{G}_1$	Can	Können	
$\mathcal{G}_2$	we	wir	vi
$\mathcal{G}_3$	afford	uns, erlauben	
$\mathcal{G}_4$	to	zu	att
$\mathcal{G}_5$	risk	riskieren	riskera
$\mathcal{G}_6$	that, kind, of	diese	den
$\mathcal{G}_7$	relationship	Beziehung	förbindelsen
$\mathcal{G}_8$	Can, we, afford	Können, wir, uns, erlauben	Har, vi, råd
$\mathcal{G}_9$	to, risk	zu, riskieren	att, riskera
$\mathcal{G}_{10}$	that, kind, of, relationship	diese, Beziehung	den, förbindelsen

	English	German	Swedish
$\mathcal{T}_1$	Can	Können	
$\mathcal{T}_2$	afford	uns, erlauben	
$\mathcal{T}_3$	to	zu	att
$\mathcal{T}_4$	risk	riskieren	riskera
$\mathcal{T}_5$	that	diese	den
$\mathcal{T}_6$	relationship	Beziehung	förbindelsen
$\mathcal{T}_7$	Can, we, afford	Können, wir, uns, erlauben	Har, vi, råd
$\mathcal{T}_8$	to, risk	zu, riskieren	att, riskera
$\mathcal{T}_9$	that, kind, of, relationship	diese, Beziehung	den, förbindelsen

Figure 4.15 – Gold and test alignment sets for hierarchical word alignment of three parallel sentences. Eight alignment units occur in both sets.

Figure 4.15 exemplifies this evaluation paradigm by means of three parallel sentences together with a gold and a test word alignment set. We anticipate here the evaluation of multilingual alignment described in Section 4.5.2, but the evaluation method based on counting successes and failures is equally suitable for bilingual alignment. In total, eight AUs can be identified that are identical in both ASs. Two AUs from the gold AS ($\mathcal{G}_2$ and $\mathcal{G}_6$) cannot be found in the test AS and one AU from the test AS ($\mathcal{T}_5$) does not exist in the gold AS. That means, we have eight true positives (TP), two false negatives (FN) and one false positive (FP).

The last case shown in Figure 4.9a, *true negatives (TN)*, refers to those AU that are not present in either AS. In medical diagnostics, for instance, the TN value indicates the number of people tested who have correctly been diagnosed as not having a particular condition; every test is an event and the total number of tests corresponds to the sum of all four event classes. With respect to word alignment, we cannot speak about events and thus the notion of correctly not identified AUs is not clear. This also relates to the fact that we cannot define true negatives in terms of two ASs (see Figure 4.9b). We can, however, calculate the number of possible one-to-one alignments as the product of the number of words for each pair of parallel sentences ($n \cdot m$) and subtract all the other countable events to obtain a substitute for the true negative events ($TN \mathrel{\widehat{=}} n \cdot m - (TP + FP + FN)$).

There is no need to know the number of true negatives to calculate *precision (P)* and *recall (R)*, two measures commonly employed to judge classification tasks.[67] Here, precision is the ratio of how many of the AUs identified by an algorithm are correct (i.e., also part of the gold AS) and recall indicates how many of the correct AUs have been found:

$$P = \frac{|\mathcal{T} \cap \mathcal{G}|}{|\mathcal{T}|} \qquad R = \frac{|\mathcal{T} \cap \mathcal{G}|}{|\mathcal{G}|} \tag{4.21}$$

As shown in Table 4.9, $|\mathcal{T}|$ can be expressed as TP + FP, $|\mathcal{G}|$ as TP + FN and the cardinality of both sets' intersection ($|\mathcal{T} \cap \mathcal{G}|$) corresponds to the number of true positives. The *F-Score* (or F-Measure) integrates both measures. An additional parameter β controls whether more emphasis is put on precision or recall. If β is not specified, F-Score typically refers to the balanced F_1-Score, which is the harmonic mean of precision and recall:

$$F_\beta = \frac{(1 + \beta^2) \cdot P \cdot R}{(\beta^2 \cdot P) + R} \qquad F_1 = \frac{2 \cdot P \cdot R}{P + R} \tag{4.22}$$

[67] In fact, Manning, Schütze et al. (1999), who explain these concepts in the context of information retrieval, argue that the number of true negatives "is huge, and dwarfs all the other numbers" and advocate the use of precision and recall.

In our example in Figure 4.15, precision is 0.889 (8/9), recall is 0.8 (8/10) and the F-Score measure yields 0.842 (16/19 = 8/9.5).

For some manual word alignments, the annotators were given two options to classify alignment links between two words: They could be marked as 'sure' if the correspondence was clear and 'possible' in not so obvious cases, for instance, for "words within idiomatic expressions and free translations and missing function words" (Och and Ney 2003). In other works (for instance, Mihalcea and Pedersen 2003), an AU was considered 'sure' only if the annotators agreed in aligning the AU in question and 'possible' if aligned by at least one annotator. Every AU in $\mathcal{G}$ is by definition possible, whereas the sure AUs form a subset thereof.[68] On that condition, precision and recall from Equation 4.21 are redefined:

$$P = \frac{|\mathcal{T} \cap \mathcal{G}_{possible}|}{|\mathcal{T}|} \qquad R = \frac{|\mathcal{T} \cap \mathcal{G}_{sure}|}{|\mathcal{G}_{sure}|} \tag{4.23}$$

The definition of precision does not change since $\mathcal{G}_{possible} \mathrel{\widehat{=}} \mathcal{G}$. Recall, in contrast, is now only calculated on the basis of the (smaller) set of sure AUs. An alignment algorithm that identifies all sure AUs but none of the possible ones will attain the maximum values for both precision and recall.

A common metric to evaluate alignment quality, the *alignment error rate (AER)*, has been proposed by (Och and Ney 2003) and since then, though criticized (Ayan and Dorr 2006; Vilar et al. 2006; Fraser and Marcu 2007; Ahrenberg 2012), used in numerous evaluations:

$$AER = 1 - \frac{|\mathcal{T} \cap \mathcal{G}_{sure}| + |\mathcal{T} \cap \mathcal{G}_{possible}|}{|\mathcal{T}| + |\mathcal{G}_{sure}|} \tag{4.24}$$

When no distinction is made between sure and possible AUs, AER becomes the inverse of the F_1, that is AER and F_1 add to 1.[69] For the sake of completeness, this relation is detailed in Appendix B.1. Null alignments are not covered by AER as the ASs are conceived to only contain one-to-one alignment links.

Most effort on the application-specific (i.e., extrinsic) evaluation of word alignment methods is directed at evaluating alignment quality in terms of the effect on statistical machine translation (SMT) systems. Common metrics for SMT are BLEU (bilingual evaluation understudy) (Papineni et al. 2002), METEOR (metric for evaluation of translation with explicit ordering) (Banerjee and Lavie 2005) and TER (translation edit rate) (Snover et al. 2006). These metrics – and others – have in common that they aim at measuring translation quality in comparison with a predetermined human translation of the same set of sentences.

[68] Tiedemann (2011) refers to the complement of sure AUs in $\mathcal{G}$ as 'fuzzy'.

[69] The AER score of our example in Figure 4.15 is 0.16.

Fraser and Marcu (2007) show that, when alignments are evaluated on a gold standard with sure and possible AUs, "AER is not a useful metric for predicting MT accuracy." They argue that in those cases "AER does not share a very important property of F-Measure, which is that unbalanced precision and recall are penalized." Ahrenberg (2012) criticizes that AER "is too coarse and does not reveal qualitative difference." Without the distinction between sure and possible AUs, AER is a balanced measure between precision and recall, which guarantees that algorithms that put significantly more emphasis on one of these will get considerably worse scores ($AER = 1$ for an empty AS and $AER \approx 1$ for an AS containing all potential AUs).

The consistent phrase error rate (CPER) (Ayan and Dorr 2006) and the translation unit error rate (TUER) (Søgaard and Kuhn 2009) use the same definition as the AER for unambiguous alignments, but the elements of their gold and test ASs are phrases and so-called translation units, which are syntactic subgraphs, respectively. By only taking into account larger units, the number of matches is expected to be lower than for counting single links between two words since the former method disregard partial matches. Both measures together convey an image of alignment quality with regard to both exactitude as to identifying larger structure and partial matches.

To see how well an alignment algorithm identifies larger AUs in comparison with single alignment links, we calculate two ratios: First, we divide the expected number of gold alignment links from the larger gold AUs by the number of AUs in the gold AS. The function Λ generates all alignment links as the Cartesian product of source and target language words. From a two-to-three alignment, we get six single alignment links, for instance. As a result, we obtain the average number of single alignment links per gold AU ($\Psi(\mathcal{G})$). In a second step, we calculate the same ratio for correct AUs, that is, the intersection of test and gold ASs ($\Psi(\mathcal{T} \cap \mathcal{G})$):

$$\Psi(\mathcal{G}) = \frac{|\Lambda(\mathcal{G})|}{|\mathcal{G}|} \qquad \Psi(\mathcal{T} \cap \mathcal{G}) = \frac{|\Lambda(\mathcal{T} \cap \mathcal{G})|}{|\mathcal{T} \cap \mathcal{G}|} \tag{4.25}$$

The Ψ function calculates the average number of alignment links per AU, here applied to both gold AUs and AUs correctly identified by the alignment algorithm (i.e., the true positives). The relation of these two ratios shows how far the actual performance of the alignment algorithm deviates from what we expect based on the gold alignments. If an algorithm is as good at identifying complete AUs as it is with regard to single links, we expect both ratios to be approximately equal. If the algorithm, on the contrary, is better in identifying single links – which we expect that any alignment algorithm is –, $\Psi(\mathcal{T} \cap \mathcal{G})$ will be greater than $\Psi(\mathcal{G})$. We name this relation alignment unit identification ratio (AUIR):

$$AUIR = \frac{\Psi(\mathcal{G})}{\Psi(\mathcal{T} \cap \mathcal{G})} \tag{4.26}$$

The closer the AUIR gets to 100 %, the better the algorithm performs in identifying whole units; lower values indicate that more AUs have only been found partially, and thus the single alignment links count as correct whereas one or more links are missing to complete some gold AUs. Since AUs define a convex hull wherein all words are aligned, these missing links must be due to unaligned words.

Table 4.10 – Frequencies and ratios for all language pairs from the example in Figure 4.15. The pair German/Swedish differs considerably from the other two pairs which is due to the small size of the example.

Language pair	$\lvert\Lambda(\mathcal{G})\rvert$	$\lvert\mathcal{G}\rvert$	$\Psi(\mathcal{G})$	$\lvert\Lambda(\mathcal{T} \cap \mathcal{G})\rvert$	$\lvert\mathcal{T} \cap \mathcal{G}\rvert$	$\Psi(\mathcal{T} \cap \mathcal{G})$	$AUIR$
English/German	34	10	3.4	30	8	3.8	0.91
English/Swedish	28	8	3.5	24	6	4.0	0.88
German/Swedish	15	8	1.9	13	6	2.2	0.87

In Table 4.10, we calculate the aforementioned ratios for all three language pairs in the example in Figure 4.15. We get similar AUIR values for all pairs, but we would anyway need larger examples to derive meaningful values. It is also possible to apply the same measure to the multilingual alignment set instead of language pairs; in that case, we also get an AUIR of 0.88 (7.0/8.0).

4.5 Multilingual Word Alignment

Multilingual sentence and word alignment share some properties; one of them is the curse of dimensionality. Even some bilingual word alignment algorithms cannot explore the whole search space of all possible alignments and need to resort to approximation or reduction of the search space (see Section 4.4.1). Multilingual word alignment differs from multilingual sentence alignment, though, primarily with regard to these characteristics:

1. Word alignments are non-monotonic, that is, AUs do not recreate the order of words in both languages. As a matter of fact, word order may vary considerably between languages.

2. Word alignments vary also in length, both in terms of characters and token count (e.g., 'colte di sorpresa'/'overtaken' in "sono state colte di sorpresa degli eventi"/"have been overtaken by events" (15/9 character ratio without blanks; 3/1 token ratio)). This variation is due to different levels of idiomaticity or to grammar (Borin 2000b).

3. While we expect to find at least partial correspondence for each sentence, the same is not true for words; there may be single words or phrases that are not expressed in our languages and thus need to remain unaligned.

4. Hierarchical multilingual word alignments require more layers to correctly represent partial correspondences.[70]

The same structure introduced for hierarchical multilingual sentence alignments on page 76 can be employed to represent hierarchical multilingual word alignments. Equation 4.1 that requires AUs to be in a subset/superset relationship if they have any element in common is revisited here:

$$\forall A_1 \in \mathcal{A}, A_2 \in \mathcal{A} : A_1 \subset A_2 \vee A_1 \supset A_2 \vee A_1 \cap A_2 = \emptyset \tag{4.27}$$

Since word alignments need to be non-monotonic, Equation 4.2 does not apply.

4.5.1 Our Approach to Multilingual Word Alignment

Triangulation is typically employed to transfer information from one language to another by means of a third one. This includes the transfer of evidence to strengthen good relations between source and language and, in this manner, improve results of the technique in question. Our prototypical approach to multilingual word alignment can be regarded as massive triangulation. We integrate evidence from multiple sources to construct binary trees of word correspondences using a hierarchical agglomerative clustering approach. Unlike our approach for multilingual sentence alignment (Section 4.3.1), where we use single-linkage clustering, we apply a variant of average-linkage clustering to multilingual word alignment to account for multiple evidence in each clustering step. It may thus be that bilingual word alignment between Bulgarian and Slovak together with a syntactic dependency relation in Swedish and a strong collocation score in Greek jointly construct a multilingual AU.

[70] We already needed more layers for manual word alignment of six languages than for manual sentence alignment of 16. We were able to cover the majority of parallel sentences with two layers.

This example reflects the two types of relation that we use: bilingual word alignment as paradigm for (direct) interlingual evidence and syntactic relations as (indirect) intralingual evidence. For every clustering decision (except for the last stage as explained below), we require evidence from at least two different sources. That way, we aim at circumnavigating errors that are well-known to occur in every probabilistic NLP application.[71]

Features

The main source for our multilingual word alignment approach are bilingual ASs generated by four word aligners introduced in Section 4.4.1:

1. Despite its age, *GIZA++* (Och and Ney 2003, see also page 111) is probably the most widely-used word aligner. It implements the IBM models 1 to 5 and an additional Hidden Markov model (HMM) (Vogel et al. 1996). Its main disadvantage is its comparably slow run-time. That is why we resort to *MGIZA++* (Gao and Vogel 2008), a re-implementation of the original alignment algorithm that takes advantage of modern multi-processor systems significantly reducing the required run-time.[72] GIZA++ as well as its multi-threaded version MGIZA++ generate asymmetric alignments.[73]

2. The *Berkeley Aligner* (Liang et al. 2006, see also page 112) employs a similar strategy. By learning the parameters of IBM models 1 and 2 and HMM models jointly from the training data, they address the issue of 'garbage collector' words, i.e. low frequent words showing a tendency to be aligned with several – non corresponding – words in the other language, that already the creators of the IBM models were aware of (Brown, V. J. Della Pietra et al. 1993).[74] Since the models are symmetric, the resulting alignments are too.[75]

[71] "two models make different types of errors that can be eliminated upon intersection" (Liang et al. 2006, on jointly training two HMMs for word alignment).

[72] Being identical with regard to the algorithms used, we shall refer to it as GIZA++, although the data has technically been processed with MGIZA++.

[73] We use 10 iterations for word class training (Och 1999) and let MGIZA++ parallelize to five threads. The alignment of 120 language pairs in both directions took more than 21 000 CPU hours (we did not measure the four additional threads separately).

[74] The authors report a significant drop of AER for jointly trained models.

[75] We use the Berkeley Aligner with the options *competitiveThresholding* and *safeConcurrency*, and let the aligner parallelize to five threads. Both options aim at increasing the alignment quality, in exchange for recall and computation time, respectively. The alignment of 120 language pairs in both directions simultaneously took more than 16 000 CPU hours (also in this case, we did not measure the four additional threads separately).

3. Dyer et al. (2013, see also page 117) propose a variant of IBM model 2 to replace the chain of IBM models used by other aligners such as GIZA++. They found that using the translation probabilities produced by model 1 as initialization of model 2 deteriorates the results and therefore use a uniform probability distribution instead. The implementation of their algorithm is called *fast_align* and generates asymmetric alignments.[76]

4. The recent word aligner *efmaral* (Östling and Tiedemann 2016, see also page 115) employs Gibbs Sampling, a variant of the Markov Chain Monte Carlo method, for sampling the probability distribution of alignments (see also Östling 2015, Section 2.5.3). Although their alignment algorithm uses a more sophisticated statistical model, it is less complex computation-wise and outperforms GIZA++ and fast_align on average, both with regard to accuracy and run-time. As it extends the IBM models with variational Bayes (see Riley and Gildea 2012), the resulting alignments are also asymmetric.[77]

We run all aligners on all parallel sentences of each language pair in both directions, except for the Berkeley Aligner, which, by reason of generating symmetric alignments, is only run once per language pair.

To symmetrize pairs of asymmetric word alignments, we subsequently perform symmetrization on the asymmetric alignments. Several methods have been suggested for that.[78] The most simple ones, *union* and *intersection* of the respective ASs, generally entail the propagation of errors (union), which leads to low precision, and the unattainability of valid n-to-m alignment (intersection), which leads to low recall. More elaborate symmetrization methods (see Och and Ney 2003, pp. 32–33; Koehn, Axelrod et al. 2005; Koehn 2010, Section 4.5.3; Tiedemann 2011, pp. 75–77; Östling 2015, Section 2.3.8.4), so-called growing heuristics, exploit the fact that multiword parts of alignments often occupy continuous positions in the sentence.[79] A method to attain symmetric word alignments directly from

[76] We use fast_align with default options. The alignment of 120 language pairs in both directions took 7500 CPU hours.

[77] We use efmaral with default options. The alignment of 120 language pairs in both directions took 3100 CPU hours.

[78] See (Koehn, Och et al. 2003, Section 4.5) for general considerations.

[79] There are always exceptions to that rule, such as particle verb prefixes in German (see Section 3.2.2). For instance 'stellt' and 'eine Notwendigkeit dar' in "In einem gemeinschaftlichen Raum ohne Binnengrenzen stellt eine Verbesserung der justitiellen Zusammenarbeit im Bereich des Strafrechts für eine effizientere Bekämpfung des organisierten Verbrechens und des Terrorismus eine Notwendigkeit dar" *'In a common borderless market, improving legal cooperation in penal law is essential to step up the fight against organized crime and terrorism.'* belong to a single AU, which corresponds to 'is essential' in English.

the translation models is described in (Matusov et al. 2004). We symmetrize alignments generated by GIZA++, fast_align and efmaral using the 'grow-diag-final-and' method. This symmetrization method is implemented by standard tools.[80]

We use the following features $\phi_y(t^1, t^2)$ for constructing edges (alignment indicators) between nodes (words, or rather tokens) of different languages (t^1 and t^2). Each feature can take numeric values from the interval $(0, 1]$, indicating a gradient degree of evidence for correspondence.

- We get most evidence on interlingual alignment edges from the bilingual word alignments that we obtain from the four aforementioned word aligners. From all four of them we get symmetric (bidirectional) alignments: ϕ_b^m (MGIZA++), ϕ_b^b (Berkeley Aligner), ϕ_b^f (fast_align) and ϕ_b^e (efmaral). If an alignment link between two tokens exists, the value of the respective feature is 1, otherwise 0. Except for the Berkeley Aligner, which inherently only generates symmetric alignments, we also count the asymmetric (unidirectional) alignments: ϕ_u^m, ϕ_u^f and ϕ_u^e. Since cases where we have asymmetric alignments for the same two tokens in both directions are neither impossible nor improbable, we divide the link count by two so that edges built from the latter features can take three values: 0, 0.5 and 1 (in case both asymmetric alignments agree).

- In addition to bilingual alignments, we compare the surface forms of the respective tokens. If they are equal, we set the feature ϕ_e to a value that starts low for short letter sequences, which are probable to be found in two languages with different connotations, but rapidly converges to 1:

$$\phi_e = 1 - \frac{1}{\frac{len(t)^2}{2} + 1} \tag{4.28}$$

  If, for instance, the word form 'casa' is found in both Italian and Spanish, ϕ_e yields 0.888 88 for this token pair. If the word form is 'antifascista', the value rises to 0.9863. The second surface form based feature, ϕ_l employs the Levenshtein distance measure, which defines the number of basic edit operations needed to convert one word into another:

$$\phi_l = \frac{l_{min} - levenshtein(t^1, t^2)}{l_{max}} - \frac{1}{2}^{\frac{l_{min}}{2}} \tag{4.29}$$

[80] For GIZA++/MGIZA++, the symmetrization tool *symal* is used; the output format of fast_align and efmaral can be symmetrized with *atools*, which is part of the fast_align software.

with $l_{min} = min(len(t^1), len(t^2))$ and $l_{max} = max(len(t^1), len(t^2))$. For equal surface forms, ϕ_l receives values close to 1. The word 'importadores' *'importers'*, for instance, is the same in Spanish and Portuguese, which results in a Levenshtein distance of 0. Since the word has 12 letters, ϕ_l thus equals $1 - 0.5^6 \approx 0.984$. The ϕ_l value decreases with increasing mismatches or fewer letters (e.g., German/Dutch 'Demokratie'/'demokratie' yields 0.869, Spanish/Polish 'momentos'/'momentach' (Levenshtein distance 3) yields 0.493 and, less similar, Spanish/Italian 'humanos'/'umani' (Levenshtein distance 3) yields 0.109). We only consider ϕ_l values of at least 0.3 as significant for alignment purposes; cases like 'humanos'/'umani' with lower values are ignored.

- Although different languages show a wide variety with regard to how corresponding units are expressed grammatically, cases in which the parts-of-speech of the respective corresponding single tokens agree are frequent given that both languages use the tagset.[81] Consider the following example in six languages:

  - English: in/ADP_1 times/NOUN_2 of/ADP_3 difficulty/NOUN_4
  - French: dans/ADP_1 les/DET moments/NOUN_2 diffiles/ADJ_4
  - German: in/ADP_1 schweren/ADJ_4 Zeiten/NOUN_2
  - Italian: nei/ADP_1 momenti/NOUN_2 di/ADP_3 difficoltà/NOUN_4
  - Slovene: v/ADP_1 težkih/ADJ_4 časih/NOUN_2
  - Spanish: en/ADP_1 épocas/NOUN_2 de/ADP_3 necesidad/NOUN_4

  Multilingual word level correspondences (AUs) are denoted by numbers. For AU 1, 2 and 3, we only see a single part-of-speech tag. AU 4 is expressed either as noun or adjective. In total, we have 48 pairwise correspondences out of which nine do not agree in part of speech. Knowing that agreement is more likely than disagreement, we can differentiate between more probable (same tag) and less probable (different tags) edges to prefer the former over the latter for the first clustering steps.

  Since function words are predominantly driven by grammar (see also Section 5.4), we define two distinct binary features: ϕ_t^c is set to 1 when two tokens of different languages possess the same part-of-speech tag and that part-of-speech tag denotes either an adjective, adverb, noun or verb (see also Section 3.2). For all other parts of speech, we set ϕ_t^f to 1.

[81] We use the first revision of universal part-of-speech tags (Petrov et al. 2012), which consists of nine universal tags.

- For FEP9, we perform syntactic dependency parsing on six languages: English, French, German, Italian, Spanish and Swedish (see Section 3.3). The resulting dependency relations are valuable clues with regard to multiword AUs. They also provide negative evidence thereof, in case the dependency label indicates that the two related tokens are not likely to form part of the same AU. For instance, subjects and predicates will typically be in separate AUs, apart from expressions such as 'there is', which, on the other hand, can easily be captured by cooccurrence measures. We define a feature ϕ_d^y for each dependency label y from the union of all respective dependency label sets that we use, which is 0 unless a dependency relation with the label in question exists. In that case, its value is 1.

- We encode the collocational strength of adjacent tokens as a single feature ϕ_c. This is motivated by the fact that multiword AUs typically appear in adjacent positions and our observation that preliminary versions of our algorithm were frequently missing single tokens in border or middle positions of larger AUs. In cases with very idiomatic phrasal correspondences, bilingual alignments often miss some involved tokens and syntactic relations do not help either, as phrasemes can be of variable length including various – if not all – types of syntactic relationships. We use the normalized simple log-likelihood measure (see Evert 2008, Section 4.2) for pairs of lemmas that occur more frequently than expected ($O > E$).

  Token pairs with high collocational strength are, for instance, 'Liu Xiaobo', 'Барак Обама' *'Barak Obama'*, 'Verbrechen gegen' *'crime against'*, 'conflictos armados' *'armed conflicts'* and 'become part'.

Weights

Pursuant to our multilingual sentence alignment approach in Section 4.3.1, we define weights w_y for each feature ϕ_y such that we get an alignment score for each possible edge by summing up the products of features and corresponding weights. The only difference to Equation 4.14 is that, here, the alignment score is calculated on token pairs and not on sentence pairs:

$$as(t^1, t^2) = \sum_i w_i \cdot \phi_i(t^1, t^2) \tag{4.30}$$

The clustering process is subdivided into individual stages detailed below. At every stage, some weights are added and/or filter heuristics are lowered. We sample the individual weights for all stages in line with the sampling for feature weights in Section 4.3.1 to determine one or more optimal configuration. Since all kinds of syntactic dependency relations are made available as features, we use evidence

from more than one bilingual word aligner and divide the clustering process into multiple stages, the search space is considerably larger compared to what we did for multilingual sentence alignment. On these grounds, we keep the division into stages and the respectively used features fix and iteratively only sample features for one individual stage each time.

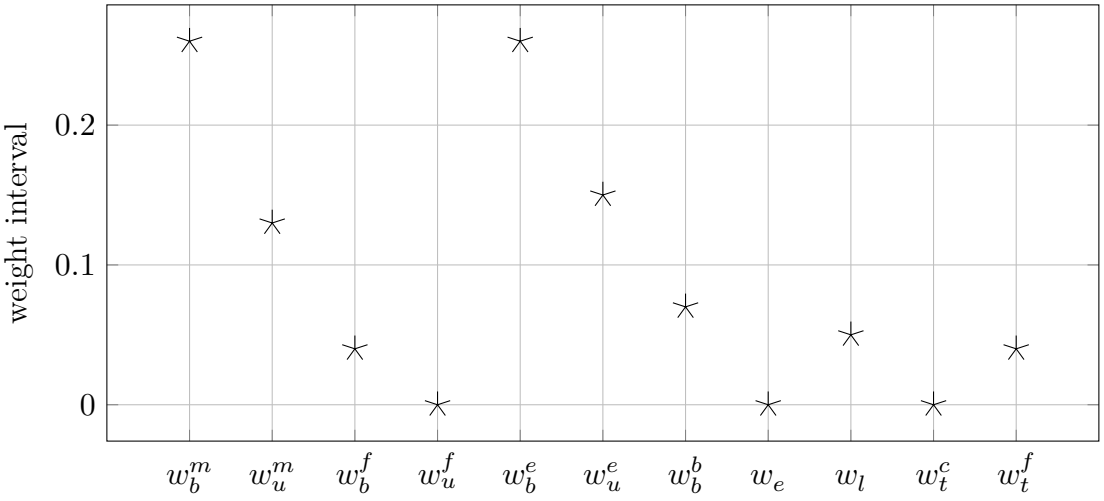


Figure 4.16 – Sampled weights for interlingual alignment features (stage 1 and 2). The weights sum up to 1 by definition.

In addition to feature-specific weights, another weighting measure we have implemented is a per language pair factor θ_{l_1,l_2}, which targets cases where different clustering options of the same alignment score are available. In these cases, we give a competitive edge to language pairs of the same language family by assigning them a θ of 1, whereas language pairs of distinct families receive a value of 0.99 and thus come second in those cases.

Stages

Separation of the clustering into consecutively executed stages allows us to prioritize the different sources of evidence and to successively raise limits with the objective of using the sources of evidence in order of decreasing reliability;[82] as opposed to the feature weights, which rank the clustering options according to the respectively set of weights.

[82] These limits are similar to the threshold for posterior decoding as explained in Section 4.4.1.

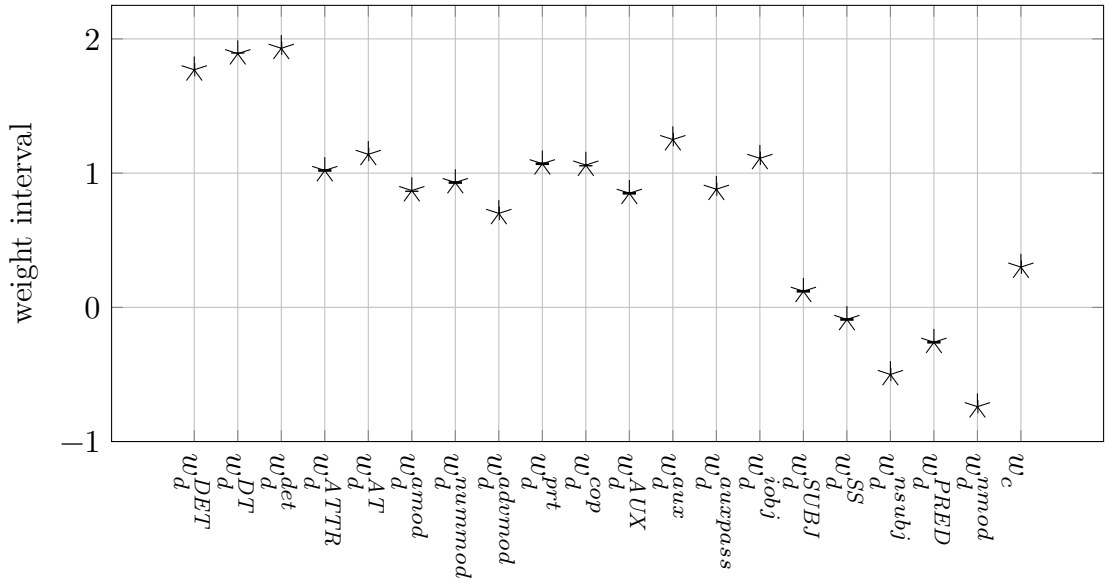


Figure 4.17 – Sampled weights for intralingual alignment features (stages 3 and 4). Different configurations lead to the same result, but variation of the respective weights is small (0.02 at most). Stars depict the most frequent configuration for this stage.

Each stage imposes limiting heuristics to keep the cluster growth controlled. A global 'degression factor' d rules out possible clustering steps that have an alignment score of less than d times the score obtained in the previous clustering step ($s \geq s_{-1} \cdot d$). We set d initially to 0.75 to subsequently lower it at later stages. This value is a compromise between being too permissive (e.g., allowing a drop to only half the previous score with 0.5) and being too restrictive (e.g., allowing only clustering with alignment scores equal to the initial one with 1.0).

In general, we prefer evidence for clustering decisions from multiple sources. Therefore, we let the heuristics require values that cannot be yielded by a single source. In the course of the progressive clustering stages, we lower these requirements.

0. A preliminary stage is dedicated to those (intralingual) syntactic relations that the bilingual word aligners typically fail to align as many-to-one or many-to-many AUs. The only feature for this stage is ϕ_d^{AVZ}, which applies to the relation between a particle verb prefix and its verb in German.[83]

[83] We are not aware of any other syntactic relation in one of the languages we have parsed that would benefit from this priority treatment. Dutch, also comprising particle verb prefixes like

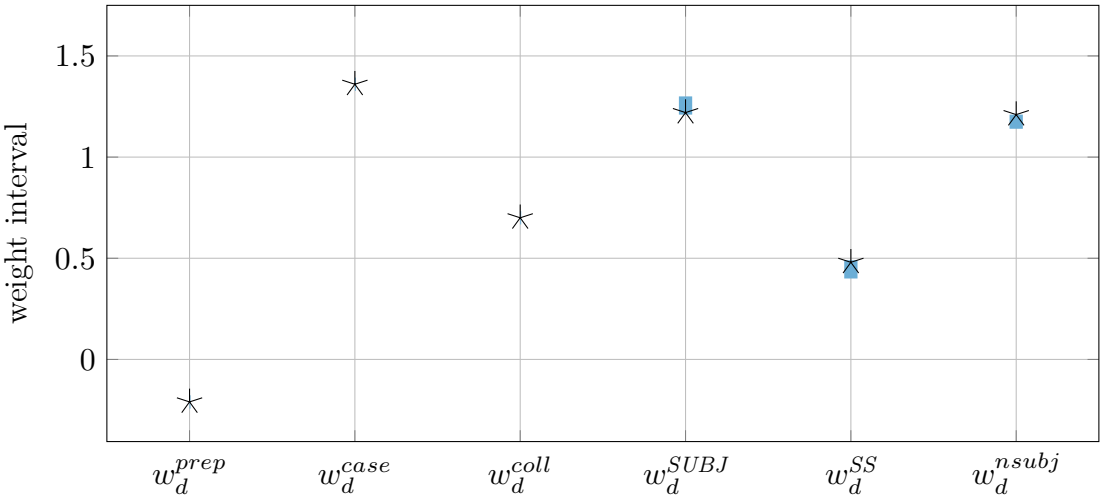


Figure 4.18 – Sampled weights for interlingual alignment features (stage 5). Some weights show a small variation. Stars depict the most frequent configuration for this stage.

Since word aligners – in particular alignment symmetrization algorithms – prefer multiword AUs of adjacent tokens (see Section 4.5.1) and particle verb prefixes typically appear at the end of a sentence in potentially long distances to their verbs (see the example in footnote 79), the prefix is often missing from the output of bilingual word aligners when applied to German. We found that the dependency parser we use (see Section 3.3) reliably links verbs and their separated prefixes so that we do not have to revert to the particle verbs identified during corpus creation (see Section 3.2.2). This is an exception to the rule of evidence from multiple sources.

1. In the first regular stage, we use all interlingual features that we have described above, specifically the symmetric and asymmetric bilingual alignments ϕ_b^y and ϕ_u^y, ϕ_e for matching surface forms, Levenshtein-distance-based ϕ_l and matching part-of-speech tags via ϕ_t. We restrict clustering steps at this stage to tokens with the same part-of-speech tag.[84] We further limit clusters to at most one token of each language. Another restriction in this stage is that a minimal alignment score of $s \geq 0.5$ needs to be reached. This

German, would be a candidate if we had parsed it and the relation between prefix particle and base verb was labelled discriminably.

[84]This restriction excludes Greek from the first stage as we did not perform part-of-speech tagging on Greek texts (see Section 3.2).

can, for instance, be the case if ϕ_b^m, ϕ_b^e and ϕ_b^b agree (see Figure 4.16). The AUs 1, 3, 5, 9, 10, 12, 13 and 15 in Figure 4.19 are generated by this stage.

2. Having homogeneous clusters with regard to part-of-speech tags after the first stage, we now lift that requirement and allow the recently created clusters to be joined with other clusters and single tokens with different part-of-speech tags. This includes tokens without tag (i.e., Greek ones). In order for a single token or cluster to be aggregated, three conditions must be met: First, there need to be at least two edges between the respective clusters, one of which can be a single token.[85] The set of edges between elements of the clusters C_1 and C_2 is given by $\theta(C_1, C_2)$; the first requirement can thus be written as $|\theta(C_1, C_2)| \geq 2$. The second requirement also concerns the number of edges, but in relation to the number of tokens in the proposed cluster, which we refer to as τ:

$$\tau = \frac{|\theta(C_1, C_2)|}{|C_1 \cup C_2|} \tag{4.31}$$

At this stage, we require $\tau \geq 0.05$, which means that there needs to be at least one edge for every 20 tokens in C_1 and C_2. This stage generates the AUs 2, 4, 11 and 14 in Figure 4.19. The last condition requires the average alignment score $\bar{s} = \sum s/|C_1 \cup C_2|$ to be at least 0.7, which prevents us from including tokens with some good alignments but little support on average over all the languages involved.

3. Clusters generated at the second stage can comprise more than one token, although their main objective is to unite first stage clusters of different part-of-speech tags (for instance clusters 11 and 14 in Figure 4.19) and single tokens that do not fit into any first stage cluster (for instance clusters 2 and 4 in Figure 4.19). The third stage, in turn, targets (intralinguistic) syntactic relations and, thus, will lead to clusters comprising two or more tokens of the same language.

   From all dependency relations available (see Section 3.3), we chose those that contribute to constituting noun phrases (determiner and modifier relations) and verb complexes (verb particle, auxiliary and copula relations), syntactic units that we frequently find in our gold word alignments (see Section 4.3.2). We also included the subject relation to account for cases like in Figure 4.26, but these relations received predominantly negative values as a result of the

[85] It is obvious that – by definition – two single tokens cannot possess more than one edge between them.

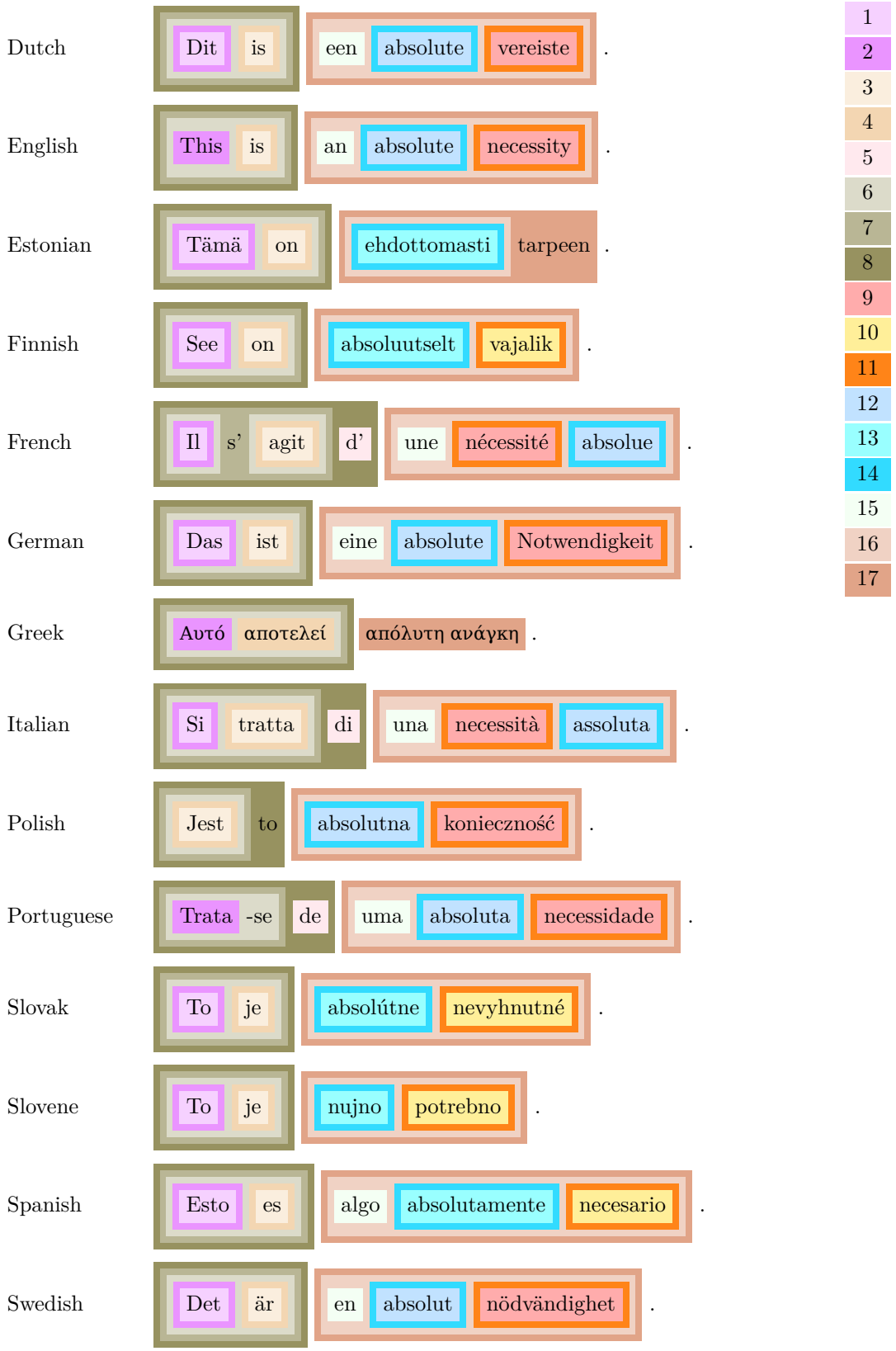


Figure 4.19 – Color-coded hierarchical AUs for a list of short parallel sentences. The three outermost boxes on the left (AUs 6 to 8) represent stages 3 to 5, the two outermost boxes on the right (AUs 16 and 17) stages 3 and 4.

feature weight optimization (see Figure 4.17), which marks them as counterproductive at this stage. The only exception is the subject relation *SUBJ* (only used for dependency relations of German), which received a small but positive value. That means that two German tokens being connected with the subject relation marginally increases their chance to be clustered.

In addition to syntactic relations, we provide the intralingual collocation feature ϕ_c at this stage. w_c receives a comparably low value of 0.3, which suffices – together with the bilingual word alignment alignments – to join the non-verbal components of expressions as shown in Figure 4.17 to the corresponding already established verb clusters.

We experimentally found that a slightly higher value of τ ($\tau \geq 0.07$), i.e. a higher number of edges required in relation to the number of elements in the respective clusters, tends to exclude a higher number of unwanted tokens from clustering. We keep the requirement of at least two edges ($|\theta(C_1, C_2)| \geq 2$), which entails that a single syntactic relation is not sufficient to establish a new cluster; there must either be intralingual support by the collocation feature,[86] a bilingual word alignment between the considered token and a member token of the existing cluster or, if an already established and not a single token is concerned, at least two syntactic edges between the two clusters.

The accumulated alignment scores can take values ≥ 1 since a complete agreement of all bilingual alignments would already result in 1,[87] and we allocate higher values for the syntactic relations as a whole in order that they dominate the remaining bilingual alignments. That is why we raised the limit of the overall score to $s \geq 1.5$. As shown in Figure 4.17, the determiner relation alone is sufficient for a token to be joined while adjective modifiers, for instance, need additional support from other features to reach that limit.

4. In a subsequent stage, we slightly lower the requirements for τ ($\tau \geq 0.06$), d ($d \geq 0.25$) and allow single-edge joins to collect the remaining, less supported tokens and smaller clusters that have not been considered for clustering at the previous step due to a comparably low alignment score (they remained below the limit given by the value of d).

[86] There can – by definition – only exist one syntactic relation between two tokens.

[87] Although, in that case, we certainly would have used the corresponding edge for clustering in one of the previous stages.

5. The last stage also aims at integrating single tokens that did not make it into a cluster in one of the previous stages. To this end, we lower τ ($\tau \geq 0.05$), d ($d = 0.1$), allow up to five different part-of-speech tags in one cluster and raise all limits with regard to the alignment score (s and $\bar{s}$). This stage frequently only generates few or no clusters.

Once all stages have been processed, we have one or more binary clustering trees with tokens as their leaves as shown in Figure 4.20. Not all tokens need to form part of a cluster; in cases where there is no correspondence for particular tokens in a single language, those tokens may remain unclustered. This is the behavior we want to accomplish, but at the same time an extra challenge as we not only need the clustering to get as far as to include all relevant tokens, but also to not include too many tokens. Agglomerative clustering without any limiting heuristics would simply result in a single cluster containing all tokens.

Transformation into Hierarchical Alignment Sets

To convert the binary cluster structure into hierarchical ASs, we solely need to identify the topmost cluster of each stage (the colored boxes in Figure 4.20) and collapse the comprised tree structure to a set of tokens. We do this for all stages except 0 and 1, which only target partial structures. The resulting structure corresponds to what is shown in Figure 4.19 without the respective innermost alignments.

For now, we focus on generating as many possibly relevant ASs as we can get from the clustering process to maximize recall. We expect that we can improve the approach by adding a heuristic to identify those clusters that correspond to the actual gold AUs and filter out intermediate ones. In some cases, we would want to exclude the topmost cluster of a particular stage if it is comprised by a superior cluster that does not include other clusters (but possibly other tokens). This is, for instance, the case for cluster #37 in Figure 4.20, which is comprised by cluster #40. That way, we would increase precision while maintaining the recall level.

4.5.2 Evaluation and Outlook

To evaluate our multilingual word alignment approach, we compare the automatically obtained AUs with the AUs in a set of **gold alignments** that we manually created for six languages: English, French, German, Italian, Slovene and Spanish.

We are primarily interested in the correct alignment of multiword AUs. Bilingual word aligners are best at aligning one-to-one correspondences,[88] followed by

[88] Which is, expectedly, the most frequent correspondence type to be found. Melamed (2001) states that "most words in a bitext translate to only one other word."

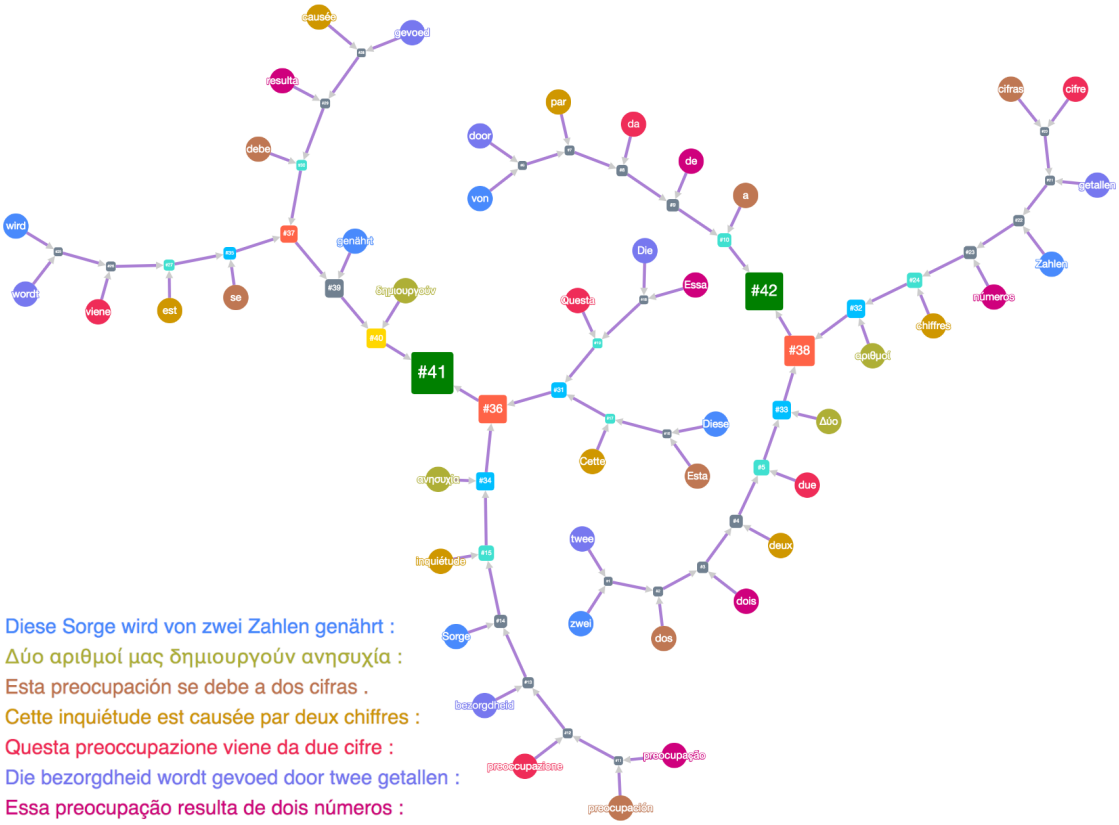


Figure 4.20 – Visualization of the resulting clusters (squares). The cluster size reflects the accumulated alignment scores. Colored clusters represent the topmost cluster generated by the respective stages: green (stage 5), yellow (stage 4), orange (stage 3), blue (stage 2), turquoise (stage 1). The algorithm's decision to join clusters #36 'this concern' and #40 'is nurtured' to cluster #41 is disputable. On the one hand, cluster #36 is just subject or object of #40. On the other hand, the expressions that correspond to 'nurture a concern' are to some extent idiomatic in some languages.

one-to-many units. Many-to-many alignments tend to pose an obstacle for those tools (see, for instance, Liang et al. 2006). In Figure 4.21, we measured the recall of the four aligners that we use in our experiments for different types of alignments (i.e., numbers of corresponding alignment elements). Starting with considerably high values for one-to-one alignments, recall drops rapidly with increasing size of the AUs. The most frequent alignment type in our gold alignments is 1:1 (78.1 %), followed by 1:2 (10.9 %), 2:2 (3.1 %) and 1:3 (2.9 %). The remaining 5 % are dis-

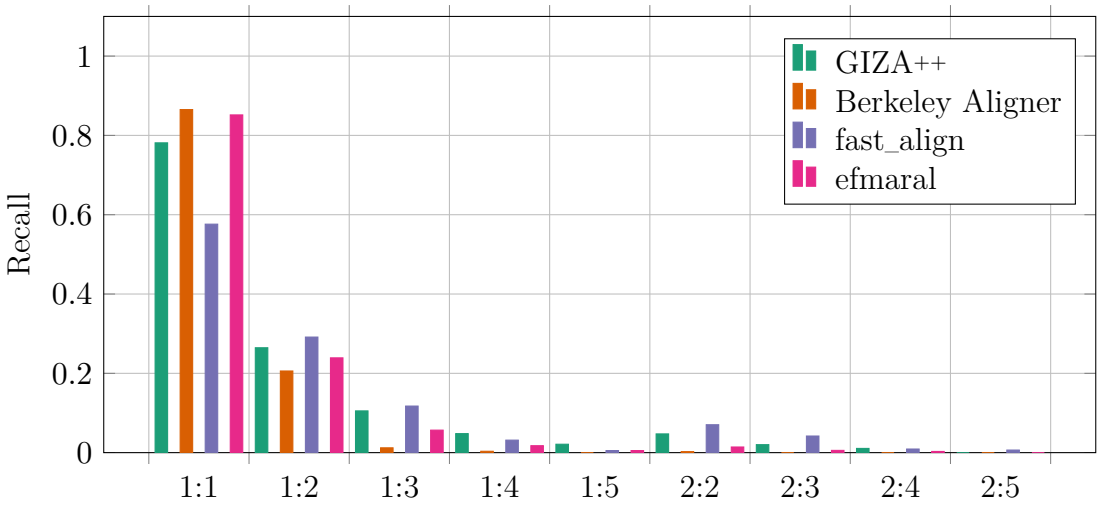


Figure 4.21 – Recall of the four word aligners with regard to different types of alignments. Pairs like 1:2 and 2:1 have been combined to 1:2. fast_align is the only aligner that also correctly identifies some 3:x alignments with $x \geq 1$, which we do not show here.

tributed over 40 other types, up to a single 9:9 correspondence.[89]

Table 4.11 – Evaluation of bilingual word aligners using our gold alignments. We compare pairwise AUs from the minimal AS of the respective language pair. Each partially correct AU counts as one false positive and false negative (a wrong AU was produced and, at the same time, the correct AU was not recognized).

	TP	FP	FN	P	R	F
GIZA++	79 599	**54 380**	43 981	**0.5941**	0.6441	0.6181
Berkeley Aligner	**86 297**	62 450	**37 283**	0.5802	**0.6983**	**0.6338**
fast_align	60 355	71 254	63 225	0.4586	0.4884	0.4730
efmaral	85 681	70 434	37 899	0.5488	0.6933	0.6127

In Table 4.11, we show performance figures for the respective aligners in comparison with all bilingual minimal ASs (see Section 4.3 on page 78) from our multilingual gold alignments. Note that minimal ASs for language pairs necessar-

[89]The idiomatic expression "throw the baby out with the bathwater" can be literally translated to French as "jeter l'enfant avec l'eau de bain" (9 tokens). In Slovene, the corresponding expression is "da se skupaj s slabimi stvarmi znebimo tudi dobrih" *'that, together with the bad things, we also do away with the good ones"* (9 tokens), which is a conceptual description of the English and French metaphoric expression.

ily cannot comprise any hierarchical AUs since the minimal number of languages per AU is two. Here, we disregard any partially matching AUs, focusing only on complete matches.

We also present F-Score figures based on single alignment links in Table 4.12 alongside the respective precision (P) and recall (R) measures, which, in comparison, give us a notion of the aligners' performance. The values calculated by counting correct links between two tokens need to be at least as high as the ones obtained by the former method since any matching AU comprises at least one matching link, but only the right set of links makes two AUs match. The difference between the two values can be regarded as an indicator of partial match frequency.

Table 4.12 – Evaluation of bilingual word aligners using our gold alignments. The units we compare are single links between tokens of the two languages, so these figures also account for partially correct AUs.

	TP	FP	FN	P	R	F
GIZA++	118 398	**25 866**	39 252	**0.8207**	0.7510	0.7843
Berkeley Aligner	125 991	33 650	31 659	0.7892	0.7992	0.7942
fast_align	112 369	38 790	45 281	0.7434	0.7128	0.7278
efmaral	**128 207**	36 956	**29 443**	0.7762	**0.8132**	**0.7943**

Comparing the figures for matching AUs and the ones for matching links, we see that in both cases GIZA++ shows the best precision, i.e. has the highest ratio of matching alignment among those identified by the aligner. The Berkeley Aligner and efmaral, on the other hand, identify the highest proportion (recall) of AUs from our gold alignments (Table 4.11). When looking at alignment links only (Table 4.12), efmaral outperforms the Berkeley Aligner with regard to recall. While the other three aligners yield similar values, fast_align does not even come close.

We also calculate the alignment unit identification rate (AUIR) described in Section 4.4.2 for each of the four aligners ($\Psi(\mathcal{G}) = 1.2757$). The Berkeley Aligner attains the best value (0.8738), closely followed by GIZA++ (0.8576) and efmaral (0.8525); fast_align scores significantly worse (0.6852).

In general, our figures are noticeably worse than the ones reported by the respective authors (Och and Ney 2003, p. 43; Liang et al. 2006, p. 109; Dyer et al. 2013, p. 647; Östling and Tiedemann 2016, p. 140). This is presumably due to the properties of our gold alignments (see below) – and we assume that the respective authors used their aligners with a configuration optimized for the respective evaluation task while we always resorted to the standard configuration.

Gold Word Alignments

We extended the gold sentence alignments described in Section 4.3.2 (hierarchical alignments of 14 892 sentences in 100 texts) by manual hierarchical word alignments. Recognizing and aligning corresponding sentences by means of identifying cognates, numbers, acronyms and matching sentence lengths amongst others is one thing, the alignment of words, phrases and other expressions is another. The decision whether two tokens directly correspond to each other or should only be aligned within a wider, more comprehensive AU is particularly challenging as the annotator is required to understand the phrasing in all respective languages, which is not the case for sentence alignment.[90] As for the alignment algorithms, the cognitive requirements on annotators for word alignment are thus considerably higher than for sentence alignment; a good command of the respective languages is necessary for this task.

Considering the language skills available in our institute, we decided to perform manual word alignment on six languages, namely English and German as representatives of the Germanic language family, French, Italian and Spanish for the Romance languages and Slovene as single Slavic language. Including a member of the Finno-Ugric language family, in our case either Estonian or Finnish, would have been interesting from a linguistics point of view as these languages are less related to the aforementioned languages than those are among themselves. Unfortunately, we could not win anybody over to the alignment task for a Finno-Ugric language. Manual alignment was performed by two student annotators working sequentially: The first annotator created complete hierarchical alignments for all languages but Slovene and the second one subsequently added Slovene while challenging the existing AUs. Problematic cases were discussed between the two of them and, if no agreement could be struck, in a larger group.[91]

Owed to the complexity of multilingual alignment and limited resources, we restricted the number of sentences to be aligned to 500, excluding primarily overly long exemplars.[92] Unlike previous work, we decided to not differentiate between 'sure' and 'possible' links[93] (see, for instance, Och and Ney 2003; Moore 2004) since

[90] A straightforward example for this are parallel support verb constructions (SVC), where the respective verbs are typically not related. The SVC 'take a walk' and its Spanish counterpart 'dar un paseo' *'give a walk'* ought to be aligned as multiword expressions with two sub-alignments, namely the determiners ('a' and 'un') and the nouns ('walk' and 'paseo'), while the verbs should remain unaligned.

[91] Unlike Melamed (1998a) (see below), we cannot calculate inter-annotator scores since our annotators performed different tasks, working on the same sentences, but with different languages assigned.

[92] In total, 816 of our gold-aligned sentences are available in these six languages.

[93] These categories are called 'exact' or 'good' and 'approximate' or 'fuzzy' translation correspondence in the Stockholm TreeAligner (Volk, Lundborg et al. 2007; Lundborg et al. 2007).

the classification of each AU into one of these two categories needs, in turn, an accurate definition of which cases are covered by which category and how to treat borderline cases.[94] Fraser and Marcu (2007) also recommend "that the Sure-only annotation style [...] be used."

We considered prealigning some tokens, primarily those that we can identify with a high confidence, to speed up the alignment process, but we did not use it. Grimes et al. (2012) report some time saving with prealignments, but they also state "A 20 % increase in speed is indeed significant, but we continue to strive for better results. We recognize that searching for and eliminating incorrect proposed alignments is also time-consuming; overhead time is required to understand each sentence and assess prealigned tokens." The reason that processing of the existing alignments adds to the time an annotator needs for processing the content of the respective sentences, the possibility that prealigning adds a bias to the manual alignment and the fact that multilingual alignment with our hierarchical alignment tool (HAT) (see screenshot and description in appendix A.2) works surprisingly fast after a short learning phase led to the decision to not perform any prealignment on the gold-aligned sentences.

The earliest documented approach of a gold standard for word alignments is, to our knowledge, the *Blinker project* (Melamed 1998a), named after the alignment tool *Blinker* ('bilingual linker'), which is a graphical user interface to perform pairwise word alignment on parallel sentences. Their 'annotation style guide' (Melamed 1998b) consist of general rules, addressing null alignments and paraphrasal translations, and an exemplified list of grammatical constellations (e.g. how to handle auxiliary verbs or repetition in conjunctions on one side). While the aim of Blinker is to create bilingual word alignments, like the ones produced by aligners such as GIZA++ later on, the *TreeAligner* (Volk, Lundborg et al. 2007; Lundborg et al. 2007) was developed to render possible the alignment of syntactic constituents, alongside the alignment of tokens.[95] Their 'alignment guidelines' for the SMULTRON treebank (Samuelsson, Volk et al. 2009) put emphasis on one criterion: that aligned parts (tokens or phrases) "can serve as translation units outside the current sentence context".

For our gold alignments, we took over this requirement as our first rule. We defined four main alignment principles[96] that we extended after a trial period with

[94] Even though defined in the alignment guidelines of the trilingual parallel treebank SMULTRON (Samuelsson and Volk 2006, 2007), the differentiation between these two categories led to some confusion among the annotators and thus inconsistencies in the annotation (Samuelsson, Volk et al. 2009).

[95] The TreeAligner can also be used for querying treebanks, but we shall focus on the aspect of manual alignment here.

[96] To keep the annotators' work focused on linguistically relevant units, we excluded punctuation from alignment.

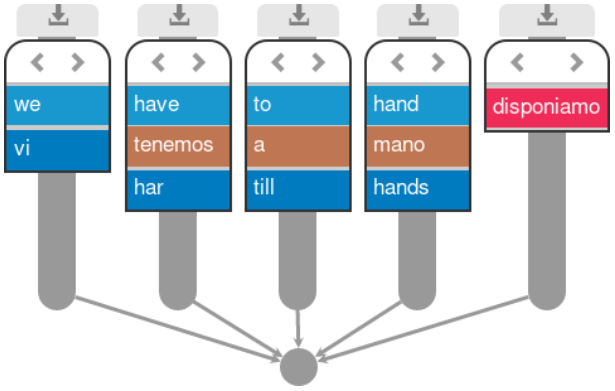


Figure 4.22 – Example representation of five AUs in four languages from our alignment tool (HAT). The rightmost set, which only contains the Italian token 'disponiamo' is not considered an AU since there is nothing at this level that 'disponiamo' aligns with. The filled circle at the bottom represents an AU that comprises the other four AUs plus the token 'disponiamo'.

best-practice examples:[97]

1. All single tokens that are considered standalone translations, i.e. translations that one would also expect to find in a dictionary without context definition, constitute a primary AU. In cases where one language uses more than one token to express the same meaning, the AU extends to all those tokens. Negation, for example, can be expressed in English with 'not', in German with 'nicht' and in French with the enclosing 'ne … pas'.

2. AUs should be minimal, thus not containing subsets of tokens in two or more languages that could be aligned separately (see Figure 4.22).[98] If a non-minimal AU is found, the respective corresponding subsets need to be separated into new AUs, which are then reattached as sub-AUs to the original AU.[99]

3. The decisive question is which sets of tokens over all available languages bear the same meaning. Grammatical categories should, consequently, play no role in word alignment. This principle also targets human annotators' temptation to look for identical structures in the respective languages.

[97] This is similar to how the guidelines for Blinker were created.

[98] Here, we make an exception for combinations of function words as their number is limited, resulting in a limited number of frequent combinations, and their informative value for linguistic tasks is lower than combinations including content words.

[99] It turned out that first creating a bigger AU with all corresponding tokens and then migrating corresponding subsets of tokens into their respective sub-AUs eases the alignment workflow.

4. Single tokens or expressions that do not correspond to any other token in the other languages should remain unaligned. We find, for example, in a German sentence 'so etwas von' in the sentence 'Davon war die Kommission so etwas von meilenweit entfernt!' *'The Commission was miles away from that.'*, a colloquial expression to intensify the following adjective, without corresponding expressions in any other language.

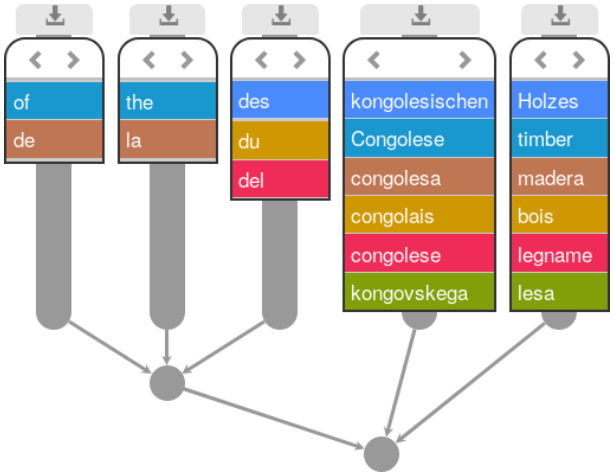


Figure 4.23 – A typical possessive AU as shown in HAT. Most languages use preposition plus article (contracted in French and Italian), German expresses definiteness and possessiveness with a definite article in genitive case with genitive suffixes for adjective and noun, and Slovene, possessing no articles, only features genitive suffixes.

These principles turned out to be sufficient to direct the alignment process. Typical language properties that require hierarchical AUs on multiple levels are:

- Compounds vs. lists of tokens: German 'Staatsoberhaupt' *'head of state'* vs. Italian 'capo di Stato', French 'chef d'État' and Slovene 'voditelj države'.

- Morphological complex forms vs. multiple tokens: German 'unwürdigsten' *'most deplorable'* vs. French 'les plus déplorables', Spanish 'más deplorables' and Slovene 'najbolj klavmih'.

- Article vs. no articles: While the other languages possess articles, Slovene does not. This frequently leads to an AU consisting of other five languages' articles, another AU consisting of six nouns and a third AU integrating the former ones.

- Grammatical cases vs. prepositions: German ‘der’ vs. English ‘of the’, Spanish and French ‘de la’. In Italian, and less frequently in French and German, prepositions and definite articles are often contracted. Here, the corresponding token ‘della’ consists of ‘de’ *‘of’* and ‘la’ *‘the’*.[100] The same applies to Slovene nouns; see Figure 4.23.

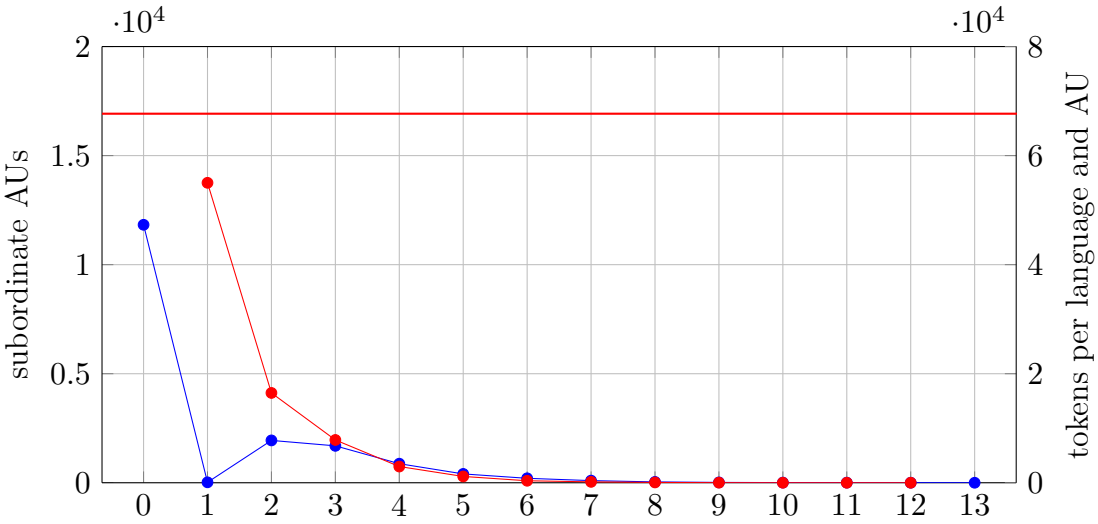


Figure 4.24 – Absolute frequencies for the number of subordinate AUs (blue) and the number of tokens per language over all AUs (red). The red line depicts the number of unique tokens.

The 500 sentences that we manually aligned in six languages yield 17 144 AUs, almost all of them comprising tokens in all six languages. All in all, 67 684 distinct tokens form part of one or more AU.[101] 69 % of the AUs are leaves of the alignment tree, i.e. they have no subordinate AU, 11 % have two, 10 % three and another 10 % four or more subordinate AUs.[102] This distribution and the number of tokens per language contained by the AUs is shown in Figure 4.24. In two third of the cases, it is a single token, in 20 % two tokens. Almost half of the AUs comprise tokens in all six languages, 20 % in five and 10 % each in four, three and two languages.

[100] The contractions were reconverted into separate tokens in the Blinker alignment task.

[101] That is about four times as many as aligned by (Melamed 1998a). If we take into account that we aligned six languages while he only did two, we still aligned a 40 % more tokens.

[102] AUs with a single subordinate AU (we only count 21 cases) are errors in the alignment structure. In these cases, both AUs contain the same tokens.

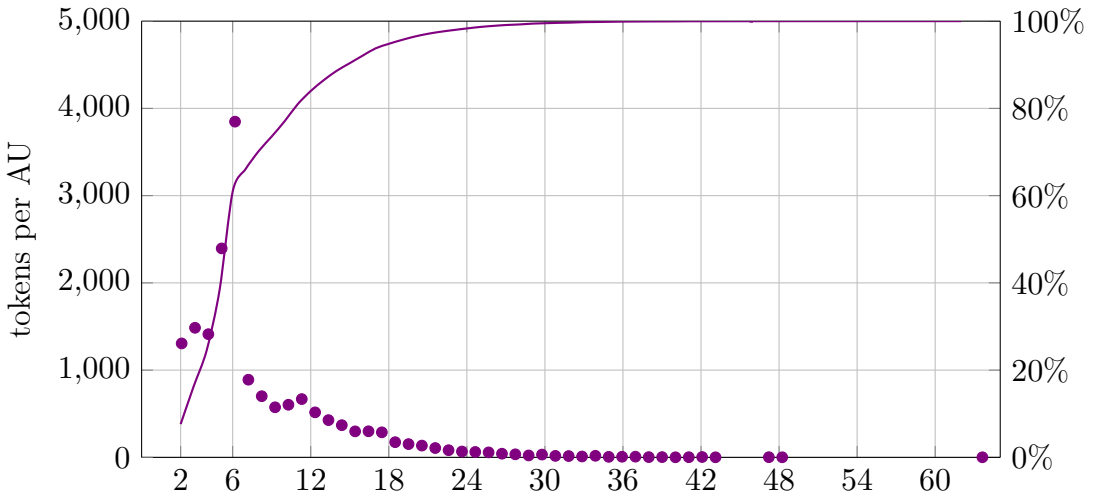


Figure 4.25 – Tokens per gold AU (violet dots). The violet line depicts the accumulated percentage of all AUs. The majority of them consist of six or fewer tokens (six tokens: 22.4 %), which in most cases corresponds to one token per language.

Regarding the size of the gold AU, we see in Figure 4.25 that AUs comprising six tokens, typically one token per language, are the biggest group, followed by five or less. The natural limit for alignment are two tokens. Some expressions like "throw the baby out with the bathwater" comprise a much larger quantity but occur considerably less frequently.

To assess the quality of our manual alignments, we randomly selected 1000 AUs from two minimal ASs: one for the language pair English/Spanish (12 286 in total) and one for the triple French/German/Italian (13 931 in total).[103] We had two human reviewers different to the annotators judge the AUs without presenting the original sentences. Their task was to decide whether each of the 1000 AUs was plausible, questionable or outright wrong. In the English/Spanish list, one AU was judged wrong and another one questionable, in the French/German/Italian list, there are six AUs judged wrong and four questionable. The most common reason for rejecting an AU were surplus tokens that should have better been aligned to a higher level in the hierarchy, to a different AU or not been aligned at all.

Results

To see how close our multilingual alignment algorithm can get to manual gold alignments we optimized its feature weights for recall. That way, we get 0.6662 as

[103]Note that the minimal AS of three languages may well contain AUs in two languages only; on condition that there is an AU comprising these two languages but not the third one in the original multilingual AS.

maximal recall value for AUs in the six languages of our gold alignments,[104] which means that we correctly identify two out of three AUs in the gold alignments. This result can be seen as similar to the recall values presented in Table 4.11 with the substantial difference that our ASs comprise six languages, which renders the identification task more complex.

In two parallel sentences, each comprising ten tokens, we have $10^2 = 100$ options for single alignment links. Assuming that these are very simple sentences and that we have ten one-to-one correspondences, then our alignment algorithm needs to identify seven of them correctly for a recall of 0.7, which amounts to a rate $7/10^2 = 0.07$. The same rate is substantially smaller when we try to achieve a recall of 0.7 in a setup of ten parallel sentences, again only looking at AUs that comprise a single word per language, namely $7/10^{10} = 0.000\,000\,000\,7$. Allowing for AUs with more than one word per language, the number of options increases and, hence, the chance to identify the correct AUs by chance decreases. This value is not reflected by precision and recall measures but hides in the false negatives (see Section 4.4.2).

Many of the missing 33 % can be credited to single alignment errors in one of the involved languages. Any missing or surplus token in one of the languages leads to rejection of an otherwise complete AU. An example of such single deviation between automatic and manual alignment is depicted in Figure 4.26 and 4.27.

The recall value does not change considerably when we remove one or more languages from the ASs, thus not converting the automatically calculated ASs on all available languages to the minimal ASs for the six languages of our gold alignments, but to subsets thereof. We observe the worst recall for the language pair French/Slovene (0.6320) and the best one for English/German (0.6917). The small difference can presumably be explained with the close relatedness of English and German. Partitioning the language subsets by whether a particular language is included or not, we also see that subsets including Slovene yield 0.6621 as lowest recall value on average and English 0.6720 as highest value. This indicates that Slovene is slightly more difficult to align than English, a fact that we would expect by reason of language relatedness. However, we incorporate syntactic relations in all other languages into the alignment process, which, although targeting the clustering process as a whole, could also raise alignment quality for those languages in particular.

Assessing pairwise alignment links between each two tokens in all identified AUs, we get approximately the same number of correct links (a recall of 0.6659), which is significantly less than the recall obtained by any of the bilingual aligners on pairwise word alignment (see Table 4.12). However, we get a similar precision

[104]The precision is at the same time very time low with 0.3574, as we have not implemented the envisaged filtering of the retrieved AUs yet.

Bulgarian	Съществуват два различни вида отношение :
Dutch	Er zijn twee benaderingen :
English	There are two different approaches :
Estonian	Olemas on kaks erinevat lähenemisviisi :
Finnish	Erämaita voidaan lähestyä kahdesta näkökulmasta :
French	Il existe deux approches différentes :
German	Dabei gibt es zwei verschiedene Ansätze :
Greek	Υπάρχουν δύο διαφορετικές προσεγγίσεις :
Italian	Esistono due approcci diversi :
Polish	Można tu wyróżnić dwa odmienne podejścia :
Portuguese	Existem duas abordagens diferentes :
Romanian	Există două abordări diferite :
Slovak	Existujú dva rôzne prístupy .
Slovene	Obstajata dva različna pristopa :
Spanish	Existen dos planteamientos distintos :
Swedish	Det finns två olika metoder :

Figure 4.26 – AU found by our algorithm. Its minimal AU for the languages used for evaluation differs in one point from the corresponding AU of our gold alignments: The German token 'Dabei' has not been included. The possible uses of 'dabei' are manifold; here it is an expression referring to the context established in the previous sentence.

(0.8020) from our multilingual AUs, which indicates that many of the false positives from the evaluation of whole AUs addressed above may be due to only small deviations from the gold alignment.

Future Options

With our multilingual word alignment algorithm identifying many gold AUs correctly, we now need to determine the characteristics of the surplus AUs returned by the conversion step that generates hierarchical ASs from the clustering trees to increase precision while maintaining recall. These characteristics can be based on inherent features such as the words included and statistics used during the clus-

Dutch	U ziet dat de antwoorden niet zo simpel zijn als ze lijken .
English	Therefore , the solutions are not as simple as that .
Estonian	Seega ei ole lahendused nii lihtsad .
Finnish	Siksi ratkaisut eivät ole niin yksinkertaisia .
French	Donc , ce n' est pas aussi sommaire que cela , les réponses .
German	So einfach sind die Lösungen also nicht .
Greek	Ως εκ τούτου , οι λύσεις δεν είναι τόσο απλές .
Italian	Quindi , le soluzioni non sono così semplici .
Polish	A zatem rozwiązania nie są takie proste .
Portuguese	Por conseguinte , as soluções não são assim tão simples quanto isso .
Slovak	Riešenie preto nie je také jednoduché .
Slovene	Zato rešitve niso tako preproste .
Spanish	Por lo tanto , las soluciones no son tan sencillas .
Swedish	Lösningarna är därför inte så enkla .

Figure 4.27 – A larger AU as identified by our algorithm. It consists of two stage 4 sub-AUs: 'as simple' and 'are not', both of which are to be found in our gold alignments. However, in the gold alignments, the second English 'as' belongs to the 'as simple' AU, so that this AU counts as false positive in our evaluation.

tering process (e.g., τ, s or s/s_{-1}) or structural configurations (e.g., the difference between an AU and the containing AU). Since we have the list of matching AUs, we can learn those properties algorithmically.

On the other hand, improving recall is also possible, although we cannot expect to find all manually created AUs. Previous experience in manual word alignment showed that inconsistencies are unavoidable (Melamed 1998a; Samuelsson, Volk et al. 2009). First of all, we have manually selected a few syntactic dependency relations. Providing the optimization algorithm with the full set of dependency relation (161 in total in all six parsed languages) will increase the time that the sampling process requires for converging, but will eventually yield weights for all relations.

We also manually defined the different stages along with their respective filtering heuristics. Automatically deriving the optimal distribution of features to stages and their filtering heuristics, yet finding the best number of stages seems a more challenging enterprise. There are possibly ways to train all these variables jointly

in a hierarchical learning approach, though this would require us to provide large amounts of training data, which we do not currently possess. A bootstrapping approach that uses our gold word alignments to align the whole corpus and use these alignments – or the best-rated parts of it – to train a model could be an option. An alternative or complement would be the manual correction of the aligned data by volunteers who could mark wrong alignments on a subset of languages they understand.

Hierarchical multilingual word alignments render possible numerous to date unrealized applications. They can be employed to answer typological questions. Structural variations may be statistically exposed and serve language learners as a reference, especially those with several L1's or L2's. Applications in computer-aided translations are also imaginable, in particular in the context where the alignments originate from.

Chapter 5

Linguistic Applications of Word Alignment

In this chapter, we describe applications based on bilingual word alignment in multiparallel corpora. They all have in common that they combine statistical evidence from multiple sources. These sources can be the same technique applied to more than one language pair or different layers of annotation and alignment.

Our first application (Section 5.1) presents a measure for **semantic relatedness** between two words represented by their lemmas. It intersects the **lemma alignment distributions** (introduced in Section 3.2.1) of two words, thus per-

Contributions

Our interface for the exploration of statistical association measures was built by Christof Bless. The prediction of learner transfer errors is joint work with Gerold Schneider. Several people have a share in Multilingwis: Simon Clematide and Martin Volk contributed to the conceptual design, implementation and testing of the first version, Dominique Sandoz built the user interface of the current version and Chantal Amrhein contributed the Credit Suisse corpus.

The author designed the database back end for Multilingwis (Section 5.2), including the query template that is used by the front end to perform corpus searches. In the same way, the author was responsible for the design and implementation of the backtranslation measures (Section 5.4); the evaluation was performed jointly. The other two sections (5.1 and 5.3) are solely the author's own work. Ongoing joint work with Gerold Schneider will rely on the overlap measure described in Section 5.1.

forming triangulation over many languages. The more these two distributions overlap, the higher we regard the probability that they represent similar, interchangeable concepts.

Since the overlap measure takes into account alignments with any other language available, we can apply it to two words of the same language or two words of different languages. As an application for lemma alignment distribution overlap on the same language, we apply the overlap measure to pairs of **German particle verbs** and their base verb, which can be interchangeable (e.g., 'steigen' and 'ansteigen' *'to rise'*) or bear a completely distinct meaning (e.g., 'lösen' *'to solve'*/*'to loosen'* and 'auslösen' *'to trigger'*/*'to provoke'*). An excerpt from our list of pairs and their respective overlap value is given in Appendix C.1.

Our example for semantic overlap between words of different languages is the identification of **false friends**, that is, word pairs from two different languages that, despite their apparent resemblance, differ in meaning. Those words that look similar on their surface (e.g., 'pregnant' and German 'prägnant' *'concise'*) have the potential to confuse language learners. A higher degree of overlap than expected for known false friends (e.g., for 'human' and German 'human' *'humane'*) can provide them with novel insights (here, that English and German 'human' are valid translations in the medical domain).

The second application, which we present in Section 5.2, is a tool for **spotting translations of multiword expressions** in several languages simultaneously. It looks up a set (or a sequence) of either word forms or lemmas in a source language (which we detect automatically) and retrieves word alignment information for every source language match in all available target languages. We aggregate statistics over the lemma sequences of all found target language sentences per language and present them to the user as **translation variants**.

Our target user group includes language learners, who benefit from the option to perform **faceted searches**, that is, to use one of the found translation variants for a subsequent search and, in so doing, continue exploring translations in the corpus. Though our search tool primarily targets non-expert users, the underlying search engine is powerful enough to perform sophisticated queries involving several layers of annotation. A future challenge will be to design a user interface that adapts to the requirements of different user groups.

In Section 5.3, we extend the notion of **cooccurrence** as described in (Evert 2008) to bilingual word alignment (Section 4.4) and **measure statistical association** thereon, aiming at the identification of word combinations that cannot be derived from the meaning of their constituents and would therefore need to be listed in a **dictionary**. We use the term **phraseme** following (Mel'čuk 1995) to refer to those combinations.

Using the example of **support verb constructions**, we detail how a low association score between the (functional) verbs of those constructions can be exploited as a measure of **idiomaticity**. To explore the properties of different statistical association measures visually, we built a web application that allows its users to define the ranking of potential support verb constructions on the basis of association measures applied to syntactic and interlingual cooccurrence dimensions.

The last application, which we present in Section 5.4, attempts to predict the **difficulty of verb- and adjective-preposition combinations** in English for language learners. Based on our prediction and general corpus frequencies, we automatically compile language-specific lists of those combinations for the purpose of supporting language learners. Combinations on these lists are presumably difficult for L2 learners of a particular L1 and at the same time frequent in English, which we estimate by frequencies obtained from our corpus.

The proposed method makes use of the lemma distribution matrix introduced in Section 3.2.1, which we multiply with the distribution of **observed foreign-language prepositions** over the entire set of a particular English verb- and adjective-preposition combination. In this way, we obtain preference values for each (correct or incorrect) preposition, which we refer to as **backtranslation score**. The ratio of each preposition's backtranslation score to the one of the correct preposition, named **backtranslation ratio**, is a measure of how probable that preposition is to be confused with the correct one. Higher scores imply a higher risk of error. Besides the compilation of lists with error-prone combinations, the backtranslation score can be used to suggest corrections in learner writing.

5.1 Overlap of Lemma Alignment Distributions as Measure for Semantic Relatedness

This section describes a method of calculating semantic relatedness between words defined by their lemmas. It makes use of alignment frequencies and is thus more reliable the higher those frequencies are. The underlying assumptions are that words with similar semantics share translations into other languages, that the size of this share correlates with their semantic overlap, and that bilingual word alignment is sufficiently reliable for calculating the required ratios.[1]

In Section 3.2, we describe how part-of-speech taggers additionally learn part-of-speech/lemma pairs from training material to not only predict part-of-speech tags but also lemmas when applied to token sequences. Furthermore, we show in Section 3.2.1 how we avail ourselves of a globally calculated lemma distribution

[1] Medeiros Caseli et al. (2010) found that "word alignment is able to attach semantic information to word and multiword units, by means of their target language counterparts."

matrix to disambiguate cases where more than one lemma got assigned to a single token. The matrix consists of the lemma alignment probabilities originally defined in Equation 3.2, revisited in Equation 5.1:

$$p_a(\lambda_t|\lambda_s) = \frac{f_a(\lambda_s, \lambda_t)}{\sum_{\lambda_{t'}} f_a(\lambda_s, \lambda_{t'})} \tag{5.1}$$

We calculate the intersection of absolute and relative alignment frequencies[2] ($f_\cap$ and $p_\cap$) for a particular lemma λ_x in a language different to the one of source and target lemma (λ_s and λ_t) by using the respective lower value:

$$f_\cap(\lambda_1, \lambda_2|\lambda_x) = min\left(f_a(\lambda_1, \lambda_x), f_a(\lambda_2, \lambda_x)\right) \tag{5.2}$$
$$p_\cap(\lambda_1, \lambda_2|\lambda_x) = min\left(p_a(\lambda_x|\lambda_1), p_a(\lambda_x|\lambda_2)\right) \tag{5.3}$$

The overlap of the lemma alignment distributions of two lemmas λ_1 and λ_2 is the weighted sum of intersecting lemma alignment probabilities for each lemma in all languages:[3]

$$O_a(\lambda_1, \lambda_2) = \frac{\sum_{\lambda_x} log(f_\cap(\lambda_1, \lambda_2|\lambda_x) + 1) \cdot p_\cap(\lambda_1, \lambda_2|\lambda_x)}{\sum_{\lambda_x} log(f_\cap(\lambda_1, \lambda_2|\lambda_x) + 1) + \epsilon} \tag{5.4}$$

We use the logarithm for absolute frequencies to account for more frequent translations without letting those dominate the overall overlap score.

The overlap measure O_a defines a weighted ratio of common translations over all other available languages. Applied to two lemmas of the same language, this corresponds to their degree of interchangeability.[4] We use the overlap measure to rank German particle verbs and their respective base verb (see also Section 3.2.2) with regard to their semantic overlap. The separable prefixes of those particle verbs modify the meaning of their base verb to very different degrees. A low O_a value (of, e.g., 'lösen' *'to solve'*/*'to loosen'* and 'auslösen' *'to trigger'*/*'to cause'*) indicates that the respective verbs of a pair have little in common with regard to

[2]We refer to the latter as alignment probabilities as they reflect the chance of a source lemma to be aligned with a target lemma given the statistics we collected from our aligned corpus.

[3]Except for Bulgarian and Greek, where tokens in our corpus do not possess lemmas and Estonian and Finnish, where the lemmas generated by the respective part-of-speech tagging model include case endings, which renders them less useful for our approach.

[4]Ignoring particular contextual requirements that would only permit one word or the other.

translations (i.e., aligned tokens in other languages) in our corpus. On the other end of the spectrum, we find pairs that bear almost the same translations, for instance, 'steigen' and 'ansteigen' *'to increase'/'to rise'*.[5]

In Appendix C.1, we list frequent particle verbs ordered by the overlap values with their respective base verbs. Comparing the beginning and the end of the list, we see that the latter pairs can often be used interchangeably (e.g., "Es reicht nicht." *"That is not enough."* vs. "Doch das reicht nicht aus." *"But these are not enough."* or "Er spart Zeit und Kosten." *"It will save time and money."* vs. "All diese Aktivitäten werden Geld einsparen." *"All this would save money."*), while the former can not. We are aware that the ranking of verb pairs cannot possibly represent a strict order of semantic relatedness, but only a rough tendency. Several factors, such as the corpus domain that our frequencies are based on, or the error rate of part-of-speech tagging and, in particular, the error rate of reattaching separated verb prefixes (see Section 3.2.2), have an influence on the final score.[6]

The same overlap measure can be applied to lemma pairs of different languages. It does – by design – not take into account the relative alignment frequencies of the second lemma given the first one and vice versa ($p_a(\lambda_2|\lambda_1)$ and $p_a(\lambda_1|\lambda_2)$), but merely calculates the indirect overlap of distributions with respect to other languages.[7] We expect the former and the latter to show a correlation in most cases, though: Two lemmas that are frequently aligned with each other (as attributes of pairwise aligned tokens) will also be frequently aligned with the same particular lemmas in other languages.

In Table 5.1, we show figures for pairs of English and German lemmas with similar surface forms, which makes them false friend candidates. To qualify as false friends, we need to show that their superficial resemblance does not come along with conforming semantics. To this end, we state, on the one hand, the conditional probabilities from the lemma distribution matrix (see Section 3.2.1), which correspond to the relative alignment frequencies between both lemmas given one or the other. In the first three examples, this probability is (almost) zero, therefore disqualifying them as mutual translations. On the other hand, we specify the alignment overlap of each lemma pair. Its value is equally low for the first three pairs, which indicates the absence of semantic relatedness.[8]

[5] Compare: "Es kann dazu führen, daß die Überschüsse der europäischen Unternehmen anwachsen, wenn die Produktivität steigt […] *"It can make the profits of European businesses grow if productivity increases […]"* vs. "[…] dass langfristig die Reallöhne und die Produktivität parallel zueinander ansteigen sollten." *"[…] that, in the long term, real wages and productivity should grow simultaneously."*.

[6] 'Abstimmen' *'to vote'* and 'auffordern' *'to ask'/'to call (up)on'* are the most frequent particle verbs in our corpus, but they are arguably less frequent in general linguistic usage.

[7] Note that O_a is a symmetric measure, while p_a is asymmetric.

[8] The resulting scores are low, but still show some overlap. We find, for instance, a single case where 'prägnante Berichte' *'concise reports'* are translated as 'pregnant reports'.

Table 5.1 – Alignment probabilities, overlap measure and overlap frequency for a selection of English/German lemma pairs with resembling surface forms. Conditional alignment probabilities that are considerably higher than the corresponding overlap measure are highlighted.

English lemma λ_e German lemma λ_g	$p_a(\lambda_g\|\lambda_e)$	$p_a(\lambda_e\|\lambda_g)$	$O_a(\lambda_e, \lambda_g)$	$\sum f_\cap$
pregnant prägnant *'concise'*	0.0044	0	0.0044	2
also also *'thus'*	0	0	0.0092	192
actually aktuell *'current'*	0	0	0.0057	148
brave brav *'well-behaved'*	0.0085	**0.5000**	0.0315	11
sensible sensibel *'sensitive'*	0.0102	0.0102	0.0606	341
eventually eventuell *'potentially'*	0.0198	0.0096	0.0683	204
sympathetic sympathisch *'likable'*	0.1139	0.2308	0.2240	109
serious seriös *'respectable'*	0.0250	**0.7045**	0.3393	1981
irritate irritieren *'confuse'/'irritate'*	0.3158	0.4474	0.4591	119
pathetic pathetisch *'lofty'*	0.1482	**0.7273**	0.5947	60
human human *'humane'*	0.0035	0.1651	0.6131	1076
automatically automatisch *'automatically'*	**0.9705**	0.5563	0.7138	4293
accept akzeptieren *'accept'*	0.5879	0.8863	0.8554	35 575
pension Pension *'boarding house'/'pension'*	0.0348	**0.9184**	0.7377	601
emotional emotional *'emotional'*	0.7665	0.7626	0.8632	1120

The following lemma pair, English 'brave' and German 'brav' *'well-behaved'*, shows a low probability for 'brav' given 'brave' in combination with a low overlap score, but a surprisingly high probability 'brave' given 'brav'. This probability is caused by annotation errors due to undetected code-switching in the German text; the German lemma 'brav' has erroneously been identified in foreign expressions ('brave new world' and 'paix des braves'). Additionally, the low overlap frequency (11 cases) renders this example less reliable.

Another remarkable lemma pair is English 'serious' and German 'seriös' *'respectable'*. The latter is translated in most cases to 'serious', the more frequent 'serious', on the other hand, aligns to six other German lemmas with a higher probability than with 'seriös'. The overlap measure, scoring at an intermediate level, indicates that other languages support this relation (e.g., French 'sérieux'), which disqualifies this pair as a false friend in the strict sense. All following lemma pairs show a larger overlap and, hence, disqualify as false friends either.

The case of 'pathetic'/'pathetisch' is similar to 'serious'/'seriös' insofar as one conditional probability is considerably higher than the other. Eight out of eleven occurrences of 'pathetisch' are aligned to 'pathetic', which shows a total of 54 occurrences. In the case of 'Pension' and 'pension', we certainly see an influence of the particular domain (plenary debates); we did not find a single case where 'Pension' would refer to a boarding house.

Table 5.2 – Alignment frequencies for the lemma 'human' in English and German with hapax legomena excluded.

no.	λ_t (λ_s = English 'human')	f_a	$p_a(\lambda_t\|\lambda_s)$
1	Menschenrecht	10638	0.6247
2	menschlich	3324	0.1952
3	Mensch	928	0.0545
⋮	⋮	⋮	⋮
10	Menschenrechtsverletzung	93	0.0055
11	**human**	**59**	**0.0035**
12	humanitär	32	0.0019

no.	λ_t (λ_s = German 'human')	f_a	$p_a(\lambda_t\|\lambda_s)$
1	humane	241	0.7651
2	**human**	**52**	**0.1651**
3	humanely	17	0.0540
4	Human	5	0.0159

A special case is the lemma 'human' in both English and German. The German lemma is aligned in most cases (77 %) with English 'humane' and, second most frequently, with 'human' (17 %). Conversely, the frequent English lemma 'human' is most frequently aligned with German 'Menschenrecht' *'human right'* (62 %), followed by nine other lemmas that are all more frequent alignments than German 'human' (0.35 %). These probabilities by themselves suggest that both lemmas 'human' are not suited as standard translations of each other although they occur as direct alignments, for the most part in expressions such as 'humane Tuberkulose' *'human tuberculosis'*, 'humane Dimension' *'human dimension'*, 'humanes Wachstum' *'human development'*, 'humane oder soziale Erwägungen' *'human or social considerations'* or 'humane Proteine' *'human proteins'*.

The overlap of these two lemmas, however, is considerably higher with 61 %. Romance languages contribute most: If we exclude non-Romance languages from the overlap measure, the O_a value raises to 0.85. The lemma alignment probability to Portuguese 'humano', for instance, is high for both English and German 'human'. The probability for English ($p_a = 0.94$) is based on 21 517 alignments of English 'human' and Portuguese 'humano'; the probability for German ($p_a = 0.91$) relies on 199 alignments. Other Portuguese lemmas that contribute marginally to the overall O_a value are 'humanidade' ($p_\cap = 0.0014$), 'humanitário' ($p_\cap = 0.0003$) and 'humanamente' ($p_\cap = 0.0001$).

The high overlap for both lemmas 'human' suggests that there is a considerably strong semantic relation between them and that we should not classify them as false friends. To address the question of what accounts for the high overlap in spite of the comparably low alignment probabilities, we look up the corpus examples where they appear together with the most supportive third language lemmas (i.e., French 'humain', Italian 'umano' and Spanish 'humano'). We see that those languages do not distinguish between English 'human' and 'humane'; both lemmas share the same single most probable alignment.

We have seen in the discussed examples that conditional alignment probabilities frequently show diverging values and, thus, do not provide a reliable view on the relation between two lemmas. The presented overlap measure performs triangulation over all available languages and is, consequently, more robust. As in the case of English and German 'human', it may result that particular lemma pairs, even though they are only used in few, specific cases, are indirectly related on a semantic level (which does not implicate that they share exactly the same meaning).

The alignment overlap measure as an indicator for semantic relatedness is restricted to single lemmas. The underlying alignment probabilities are calculated based on optimal alignments, that is, a one-to-one alignment supported by all four aligners that we use (see Section 4.5.1). If a single word corresponds to a multi-

word expression consisting of, for instance, two words and all four aligners correctly identify this one-to-two correspondence, the probability mass of the alignment distribution for the source lemma is spent partly on the first and partly on the second lemma of the target multiword expression.[9]

For the overlap measure to account for such a relationship, both alignments to the third language need to match. Otherwise, if we miss, for instance, one of the two words on one of the two correspondences, the overlap score will only incorporate the partial overlapping probability of the other word. In case one of the two lemmas that we compare itself forms part of a multiword expression, assuming that the aligners correctly handle the one-to-many correspondences, we will see a considerably lower overlap score compared to single word correspondences. The overlap measure is, thus, only reliably applicable to single word pairs.[10]

Regarding the reliability of the resulting overlap score, raw frequencies also need to be considered to estimate the influence of annotation and alignment errors. A high overlap that is supported by most of the available languages[11] is evidently more reliable than a high score produced by only a few third languages.[12] A low number of actual triangulation points can also negatively influence the reliability of the overlap measure. However, a low absolute overlap frequency is typically accompanied by a low number of supporting languages. The two lemma pairs in Table 5.1 with distinctive low frequencies, 'pregnant'/'prägnant' and 'brave'/'brav', are supported by only two and five languages, respectively.

In future works, this method can be used to support language learners in three ways: First, we can identify false friends in learner texts by looking up potential false friends in every language the learner knows, and try if the other words' standard translations would fit better collocation-wise, which indicates a potential transfer error. Second, applying the method to each two superficially similar looking words of a language pair, we can automatically obtain corpus-driven lists of false friends that serve as didactic material for language teaching, even for rare combinations of languages. Third, the entire set of overlapping and disjoint lemma alignment probabilities of two words in one language can support a learner's decision in writing tasks by helping her distinguish the semantics of two seeming synonyms, for instance, between a particle verb and its corresponding base verb in German (see above).[13]

[9] This is due to the requirement that alignment probabilities for a given lemma sum up to 1.

[10] The highest scoring percentile of frequent lemma pairs for English/German consists for the most part of month and country names, but we also see some other parts of speech (e.g., 'between', 'humanitarian', 'and', 'three', '%').

[11] We say that a language supports the overlap score if there is at least one lemma of this language that has a positive alignment probability with both lemmas in question.

[12] We found eight supporting languages out of ten a reasonable lower limit.

[13] For the lemmas 'liberty' and 'freedom', we get an overlap score of 0.88.

So far, we have calculated the alignment overlap for some lemma pairs that we considered interesting. We plan to systematically calculate it for all lemma pairs with high surface similarities, which can be based on the Levenshtein distance or similar distance metrics. Another option is to translate the respective lemmas of different languages to their phonetic transcription and subsequently compare these transcriptions between languages to find pairs that are pronounced similarly while potentially having dissimilar surface forms (e.g., French 'gâteaux' *'cake'* pronounced /gɑ.to/ and Spanish 'gato' *'cat'* pronounced /ˈga.to/). This conversion will also allow us to calculate phonetic similarities of false friend candidates with a high surface similarity (e.g., 'slut', which is pronounced /slʌt/ in English and /slʉːt/ in Swedish, meaning 'end' or 'over').

5.2 Multilingual Translation Spotting

Single words typically bear a meaning.[14] When two or more words together bear a meaning, we refer to them as *multiword expressions* (MWE), irrespective of whether this meaning is compositional, that is, it can be derived from the meaning of its components, or idiomatic, that is, the meaning is non-compositional or the composition is not transparent to the typical language user. We consider the term 'multiword expression' the most neutral in comparison with 'phraseme', 'idiom' or 'idiomatic expression' and 'formulaic sequence'. It encompasses all the other ones and can be applied to ordinary nominal compounds as well as set phrases.[15]

If we want to know how a single word translates to another language, we can simply look it up in the lemma alignment distribution matrix (see Section 3.2.1). A lookup for English 'human' into French yields a Zipfian distribution (Zipf 1949): 'humain' *'human'* (89 %), 'homme' *'man'*/*'human being'* (10 %), 'humanité' *'humanity'*/*'humankind'* (0.5 %) and other, less frequent translations, including several alignment hapax legomena with a high probability of being false positives. For multiword expressions, however, there is no similarly easy statistics that one could consult to learn about translation variants of a given expression.

This is where we put forth our tool for online searches of translation variants in multiparallel corpora *Multilingwis (multilingual word information system)* (Clematide et al. 2016; Graën, Clematide and Volk 2016; Graën, Sandoz et al. 2017).[16] We use the term translation variants to refer to distinct lemma tuples that are aligned with a list of search terms found in a corpus. All translation

[14] Notable exceptions are words that bore a meaning in the past and nowadays only exist in fixed expressions (e.g., multiword adverbs (Volk and Graën 2017) in German 'klipp und klar' *'in plain language'*, 'fix und fertig' *'completely exhausted'* or English 'to and fro').

[15] *Perish the thought!*

[16] https://pub.cl.uzh.ch/purl/multilingwis

variants are translation equivalents of the source language terms, but we want to stress their character as alternative translations here. Since translation variants are tuples, the order of their elements matters. The lemma sequences "ancien tradition" and "tradition ancien" are thus two different French translation variants of the English expression "old tradition".

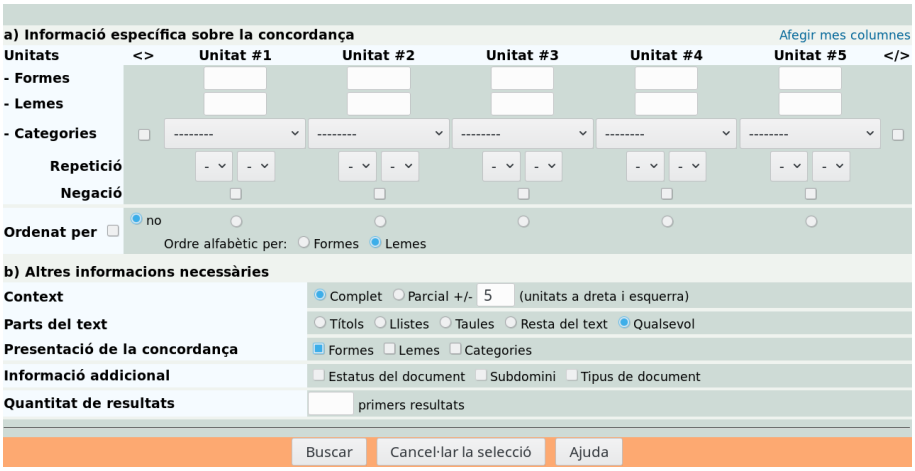


Figure 5.1 – Corpus search form for sequences of words in BwanaNet.

In contrast to well-known corpus query tools such as CQPweb (Hardie 2012) or ANNIS (Chiarcos, Dipper et al. 2008; Krause and Zeldes 2014), we want to offer a corpus search tool to non-expert users such as translators, terminologists and language learners. In line with existing online search tools for parallel corpora,[17] but opposed to, for instance, the CQP-driven BwanaNet corpus search interface[18] in Figure 5.1, we provide the user with a simple input form (Figure 5.2).

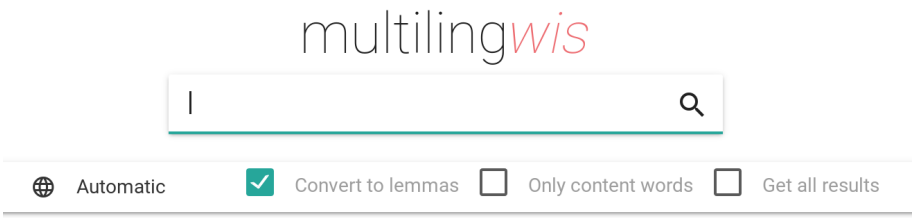


Figure 5.2 – Minimalistic search form in Multilingwis (second edition).

Multilingwis performs a corpus search on the list of search terms entered by the user and shows frequencies of translation variants in all available languages and a list of corpus examples, which the user can inspect individually. If the

[17] We compare Glosbe, Linguee and Tradooit in (Volk, Graën and Callegaro 2014, Section 4.1).
[18] http://bwananet.iula.upf.edu/

list of search terms corresponds to a frequent multiword expression, we expect the frequency lists to show a Zipfian distribution of translation variants, just like the lemma alignment distribution for single words. Figure 5.3 shows query form, frequency lists and example view in Multilingwis.[19]

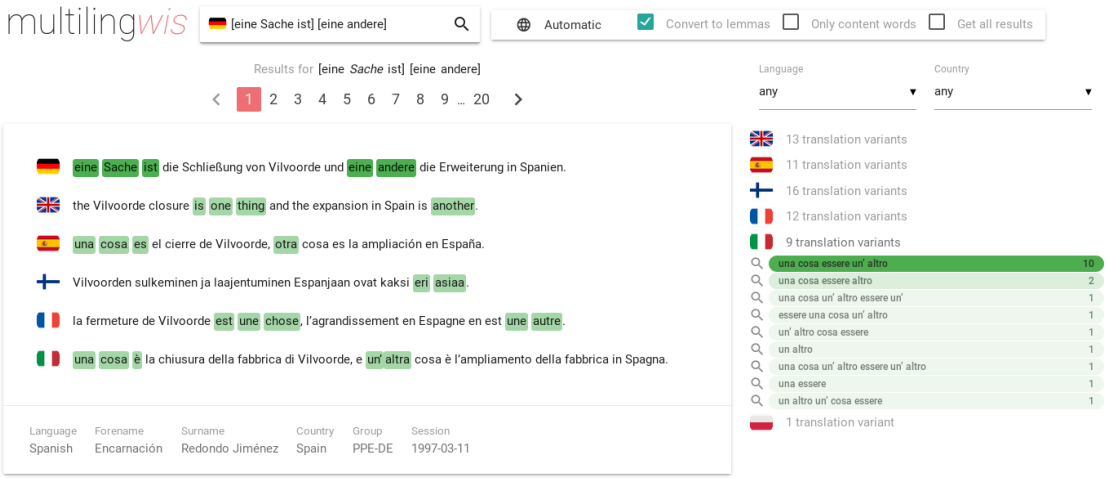


Figure 5.3 – The Multilingwis graphical user interface displays frequency lists of translation variants to the right and corpus examples to the left. The user can confine the corpus examples by choosing particular translation variants.

We automatically recognize the language of the search terms entered. To this end, we look up each term in the list of word forms in the respective corpus together with their frequencies. For multiword expressions with three or more words, there is typically only one language that features all the terms as word forms. If one of the terms does not exist in any language, we immediately abort the search process. Single terms or, considerably less frequently, pairs of search terms can be ambiguous. In this case, we construe the language with higher frequencies as intended. This is why in our corpus (FEP6), for instance, 'niego' as word form of the Polish personal pronoun 'on' is preferred to the Spanish verb 'negar' *'negate'* and 'kitten' is interpreted as the German verb 'kitten' *'to cement'*/*'to repair'* and not as a young cat in English. The expression 'con calma' *'calmly'* exists in both Italian and Spanish; in Italian, 'calma' can be noun or adjective. Though 'calma' is slightly more frequent in Italian, 'con' is observed almost twice as often in Spanish. On these grounds, Multilingwis decides to perform a search in Spanish. Nonetheless, the user can always overrule Multilingwis' choice.

[19] We shall only describe the most recent version of Multilingwis (Graën, Sandoz et al. 2017) as it further develops the techniques implemented in the first version (Clematide et al. 2016; Graën, Clematide and Volk 2016).

Once the language is set, we perform a corpus search in two steps: The first step identifies sentences in the source language that match the search terms together with the matching tokens, which we refer to as *(search) hits*. The second step uses word alignment information to retrieve the corresponding tokens in all other languages, construct the translation variant as tuple of their lemmas and calculates frequency lists on the entire result set.

Besides the source language, the identification of search hits is controlled by the three parameters shown in Figure 5.2.[20] By default, we perform a search on lemmas to account for morphological variation. When disabled, the respective word forms of the tokens have to match the search terms. The second option, the limitation to content words,[21] affects both the search term list and the query. When enabled, function words are skipped, which in some cases leads to the inclusion of alternative expressions (e.g., "human rights violations" when "violations of human rights" is searched for). The last option disables a limit for the number of hits. We consider approximately 1000 hits sufficient to derive meaningful statistics of translation variants. Furthermore, we do not expect the user to page through several hundred results. If a search exceeds this limit, Multilingwis advises the user who, in turn, can rerun the search with a significantly higher limit.[22]

The second step is to look up word alignments for all tokens of each hit. As long as at least one aligned token in a particular target language is found, we include the resulting token tuple (possibly a 1-tuple) as lemma variant in the frequency list and register the combination of hit tuple and translation tuple as example for that variant. The respective languages are treated independently; we have one source and several target languages. However, example sentences are registered with their respective translation variants in all languages. That way, the user is able to find multilingual examples for a combination of particular translation variants in several languages.

In Multilingwis, the search terms are logically combined in a way to which users are accustomed from web search and document retrieval systems: Only sentences matching all the search terms are retrieved. To further narrow down an expression, we support phrasal restrictions, that is, the requirement of two terms being adjacent to each other or holding a particular token distance by means of placeholders.[23] Unlike popular web search engines, we use square brackets and

[20] More filters can be configured on the basis of metadata provided for each sentence.

[21] We define them in terms of universal part-of-speech tags (Petrov et al. 2012) as nouns, verbs, adjectives and adverbs, everything else is regarded a function word. This definition, though arguably imprecise, suffices for the purpose of non-expert corpus search.

[22] For technical reasons (memory, bandwidth, etc.), we always limit the number of hits to 100 000.

[23] This is how the first version of Multilingwis interprets a sequence of search terms. That version also always performed a lemma-based search and consistently ignored all function words.

not quotation marks to denote phrases. The query ⟨`[tradition ancien]`⟩, for instance, requires the terms 'tradition' to be immediately followed by 'ancient'. We consider an opening and a closing bracket more intuitive if there is more than one phrase. Moreover, we support placeholders inside of phrases. A lemma-based search for the query ⟨`[keep * head above water]`⟩, for instance, yields "keeping their heads above water", "keep our heads above water" or "keep his head above water".

Another feature we consider important for exploring multiword expressions is to facilitate backward searches on the identified translation variants to see how these get translated. Since we collect those variants disregarding the positions of the respective tokens, we cannot apply phrasal restrictions to those searches. This kind of faceted search allows the user to investigate difference in meaning and the contexts that translation variants for the initial search terms translate back to.

Technical Details

We provide Multilingwis as a query tool with few technical dependencies. Besides technological requirements (a PostgreSQL database and a web server with scripting support), Multilingwis needs a part-of-speech-tagged, lemmatized and word-aligned parallel corpus in a CoNLL-like format.[24] Parting from this tabular input data, which is uploaded to a temporary database table as a first step, the hierarchical corpus structure is derived and the attributes (word form, lemma, part-of-speech tag, etc.) and relations (one-to-one word alignments) are normalized. Word alignments are regarded as symmetric in the database. If they are not symmetric in the input file, union symmetrization is performed automatically.

As a test case we exported the relevant parts of our own corpus database FEP6, which comprises the European Parliament's debates (see Section 3) to import it into Multilingwis. This is a straightforward task as Multilingwis' database schema is modeled on structures that we successfully use in our corpus database. FEP6 covers the languages English, Finnish, French, German, Italian, Polish and Spanish, which is also depicted in Figure 5.3. The second corpus we successfully imported into Multilingwis is the *Credit Suisse Bulletin corpus* (Volk, Amrhein et al. 2016), a parallel corpus comprising articles from the Credit Suisse Bulletin over a period of more than a hundred years in up to four languages: English, French, German and Italian. The imported subset comprises more than two million tokens from recent issues in English, French and German and 1.3 million tokens in English.

[24] We explain the corpus format, the import and the configuration of the web interface in detail in Multilingwis' software repository: https://gitlab.cl.uzh.ch/sparcling/multilingwis2/

All attributes that are needed by the search are indexed with regular B-tree indices. For those relations that are traversed for accessing other attributes, we use composite indices so that the required values can be retrieved directly from the index. The word forms and lemmas of each sentence's tokens are indexed by means of an inverted positional index, which allows us to exclude all sentences that do not contain all the required search terms in the first place. In case phrasal restrictions are specified, we map those to the 'FOLLOWED BY' operator that has recently been added to PostgreSQL full text search capacity (Bartunov and Zakirov 2016; PostgreSQL Global Development Group 2017).

Outlook

The CoNLL-like import format used by Multilingwis already comprises many features that are typically represented in a text corpus, except for word alignment, which is essential for Multilingwis to work. We currently do not make use of the language-specific fine-grained part-of-speech tags, morphological information and dependency relations. Storing these attributes, indexing them and making them available in the form of a more sophisticated search term notation is thus arguably feasible. The challenge, however, is to bear in mind the trade-off between an easy-to-use interface and a full-fledged corpus query language. It may be advisable to provide graded variants of user input options to take account of different user groups' requirements.

5.3 Phraseme Identification

The previous section (5.2) deals with finding translation variants for multiword expressions specified by a user. Multiword expressions are understood here as any set of word forms or lemmas that appear together in a source language sentence. The corresponding (i.e., word-aligned) tokens in the target languages are denoted translation variants. Their frequency distribution together with the total number of search hits often indicates whether the given list of search terms forms a linguistic unit or if those terms are about the same topic.[25] In this section, we explore association measures to identify phrasemes, lexically restricted combinations of words.

[25]Searching for sentences with the terms 'patient' and 'doctor', we find translation variants with the corresponding lemmas of both English words in all languages (e.g., (médico, paciente) and (paciente, médico) in Spanish). Both possible orders are frequent, which suggests that the terms merely belong to the same topic but do not form a linguistically relevant unit together.

For Mel'čuk (1995), phrasemes are those combinations of at least two words that need to be comprised and explained by a dictionary. That implies that a language user cannot derive the phrasemes' meaning from its parts. A 'blue background' is arguably a background that is blue, but 'blue card' (concrete noun) or 'blue murder' (abstract noun) are concepts that need to be explained in a dictionary. The challenge now is to automatically distinguish between mere compositional and phrasemic combinations. A clear division between the two is inherently difficult due to expressions that can either denominate real-word concepts or be used figuratively (e.g., 'black box', 'red tape', 'green light', 'white paper').[26]

A momentous observation made by (Firth 1957a; Firth 1957b) is that particular words impose restrictions on others. Though he was arguably not the first person to observe this interrelation, he is known for describing it and coining the now widely-used term *collocation* (Firth 1957b, p. 195) for those restrictions. By way of example, he lists typical adjectives that can be collocated with the noun 'ass'. While the respective combinations bear particular meanings (e.g., 'silly ass'), his concept of collocation is based on "mutual expectancy" of the involved words, "the hearing, reading or saying of [the collocation]" and not its meaning (ibid.). Evert's (2004) definition of collocations accords with Mel'čuk's (1995) definition of phrasemes, that is, that the respective combinations need to be listed in a dictionary by reason of not being transparent to a language user. In his work, he focuses on the frequency of those combinations, which can be obtained from sufficiently large corpora[27] and provides evidence for their collocational status.

Statistical Association Measures

Several statistical association measures have been suggested to identify collocations. (Evert 2008, Chapters 4 and 5) gives an overview and describes the respective measures' strengths and shortcomings. In particular, he highlights the tendency of some measures to overrate particular cases, for instance, when one word of a pair is observed only few times in a corpus, and points out the commonly applied frequency threshold to countervail this issue. He also classifies association measures into simple and sophisticated association measures.

Simple association measures are those based on the observed cooccurrence frequency (O) of a word pair and the expected cooccurrence frequency (E) of that pair under the assumption that both words in question are independent of each other,

[26] We have chosen to exemplify those combinations with color adjectives modifying concrete nouns because they stick out statistically in our corpus, where they are predominantly used in a figurative sense. Apart from those ambiguous combinations, we find, for instance, 'bottomless pit', 'blind eye' or 'blank cheque'. In German, typical exemplars are 'runder Tisch' *'round table'*, 'erster Schritt' *'first step'* and 'schmaler Grat' *'thin line'* (literally *'narrow ridge'*).

[27] A means that Firth did not have at his disposal.

that is, equally distributed in the corpus. The sophisticated association measures, in contrast, also take into account the frequencies of all other possible events. All four possible events can be represented as a contingency table (Table 5.3), which we borrow from (ibid.).[28]

Table 5.3 – Contingency table for the observed frequencies of words w_1 and w_2. O_{11} corresponds to the O value of the simple association measures.

	w_2	$\neg w_2$
w_1	O_{11}	O_{12}
$\neg w_1$	O_{21}	O_{22}

The expected frequencies are the result of multiplying the independent frequencies of both respective events (word w_1 or $\neg w_1$ and w_2 or $\neg w_2$) in relation to corpus size N (Table 5.4, also borrowed from (ibid.)).

Table 5.4 – Contingency table for the expected frequencies of words w_1 and w_2. E_{11} corresponds to the E value of the simple association measures. In each cell, the absolute frequencies of both independent events (e.g., $f(w_1) = O_{11} + O_{12}$) are multiplied and divided by the sample size N (i.e., the number of sentences in the corpus).

	w_2	$\neg w_2$
w_1	E_{11} $= (O_{11} + O_{12}) \cdot (O_{11} + O_{21})/N$	E_{12} $= (O_{11} + O_{12}) \cdot (O_{12} + O_{22})/N$
$\neg w_1$	E_{21} $= (O_{21} + O_{22}) \cdot (O_{11} + O_{21})/N$	E_{22} $= (O_{21} + O_{22}) \cdot (O_{12} + O_{22})/N$

We calculate observed and expected frequencies for all pairs of consecutive tokens in our corpus (FEP9) based on both word forms and lemmas. To demonstrate the respective simple association measures, we extract those pairs where the second lemma is 'fish', independent of its part of speech, and rank all pairs according to six different association scores (ibid., p. 1225). In Table 5.5, we list the overall best scoring potential collocations and the ranks for the respective measures. Note that some measures (local mutual information (local-MI), t-score and simple log-

[28] We shall stick to the simple association measures since they "often give close approximations to the more sophisticated association measures" and "[t]herefore […] are sufficient for many applications" (Evert 2008).

Table 5.5 – The 30 best scoring potential lemma collocations for $w_2 = \textit{fish}$ based on the n-best of six different association scores. The r values define the corresponding ranks of the following association measures: (1) mutual information (MI), (2) MI^2, (3) local mutual information (local-MI), (4) z-score, (5) t-score, (6) simple log-likelihood (simple-ll) (Evert 2008, p. 1225). $f(w_2)$ is constantly 3383 and therefore omitted.

w_1	w_2	$f(w_1)$	O_{11}	E_{11}	r_1	r_2	r_3	r_4	r_5	r_6
juvenile	fish	176	30	0.014	9	1	2	1	4	2
immature	fish	36	13	0.003	3	2	8	2	23	6
white	fish	585	28	0.046	16	8	3	8	5	3
of	fish	1 421 044	575	111.407	116	20	1	24	1	1
conserve	fish	263	24	0.021	13	7	5	7	6	4
deep-sea	fish	101	17	0.008	10	5	6	5	15	5
undersized	fish	16	8	0.001	2	3	19	3	34	16
scorpion	fish	5	4	0.000	1	4	36	4	65	30
edible	fish	23	8	0.002	4	6	20	6	35	17
migratory	fish	403	17	0.032	17	12	9	12	16	8
fresh	fish	1275	21	0.100	29	17	7	17	11	7
wild	fish	389	16	0.030	18	14	12	14	18	11
vessel	fish	2328	22	0.183	37	22	10	21	10	9
diseased	fish	21	5	0.002	6	9	30	9	53	26
freshwater	fish	24	5	0.002	7	10	32	10	54	28
to	fish	1 316 995	224	103.250	156	43	4	70	2	10
preserve	fish	2452	20	0.192	40	24	13	23	13	12
tuna	fish	726	14	0.057	26	19	16	19	21	15
demersal	fish	28	5	0.002	8	11	34	11	55	29
catch	fish	2201	18	0.173	39	25	15	25	14	13
dwindle	fish	119	7	0.009	15	16	25	16	41	23
young	fish	8348	24	0.654	66	32	14	32	8	14
deplete	fish	359	9	0.028	23	21	23	20	31	22
canned	fish	50	5	0.004	12	15	38	15	56	32
landed	fish	6	2	0.000	5	13	63	13	101	55
on	fish	404 469	95	31.710	140	57	11	72	3	18
farm	fish	2176	14	0.171	47	28	21	28	22	21
few	fish	16 705	25	1.310	83	45	17	46	9	19
protect	fish	11 435	22	0.896	75	40	18	41	12	20
fleet	fish	2084	11	0.163	51	34	26	34	28	24

likelihood (simple-ll)) assign high scores to pairs with frequent prepositions ('of', 'to', 'on'), while, in particular, the mutual information measure (MI) ranks those pairs considerably lower.[29]

From our point of view, few of the best scoring collocation candidates with 'fish' qualify for a dictionary entry; the best candidates are compounds ('scorpion fish', 'deep-sea fish' and 'freshwater fish'). At least, most of the adjective + noun combinations can be understood if both adjective and noun are known to the language user. There are, nonetheless, combinations that are restricted by language. For instance, 'juvenile' bears a meaning similar to 'young', which is also represented, but is limited to animate beings, whose life follows a predetermined cycle.[30]

Further investigating the MI measure, we see that the 200-best list predominantly contains proper nouns with low frequencies (from two to five occurrences). Among them, we find organizations (e.g., 'Shin Bet', 'Nueva Izquierda', 'Shining Path', 'Bayern München'), persons (e.g., 'Ayn Rand', 'Hannah Arendt', 'Bertrand Russell', 'Vargas Llosa'), places (e.g., 'Yad Vashem', 'Kloster Andech') and companies (e.g., 'Hewlett Packard', 'SmithKline Beecham'). Exceptions are technical terms (e.g., 'staphylococcus aureus', 'mucous membrane', 'beta interferon', 'light-emitting diode', 'Simpler Legislation') and common nouns such as 'roller coaster', 'spiny dogfish' or 'oily slime'. Figure 5.4 shows observed frequencies of 200-best lists for six simple association measures. Their division into two groups by frequency is striking. The best scoring potential collocation of the three measures (local-MI, t-score and simple log-likelihood) that prefer higher observed frequencies is 'of the' with $O = 441\,459$.

So far, we have only based our calculation of the cooccurrence frequency O on consecutive token pairs. If we loosen this restriction and say that words cooccur if they are to be found within a particular distance.[31] we will also identify expressions that typically embrace other words as potential collations (e.g., French negations 'ne … pas', 'ne … plus', 'ne … jamais', 'ne … rien'). Since we base the frequencies in this case on token sequences and not on any deeper linguistic properties, Evert (2008) refers to it as *surface cooccurrence*. Measuring cooccurrence on consecutive token pairs is thus a special case of surface cooccurrence.

Another option that yields even higher frequencies is to count pairs of words that cooccur in the same sentence. This *textual cooccurrence* (ibid.) yields words that have a semantic relation insofar as they are frequently used together in the

[29] It has, on the other hand, "a tendency to assign inflated scores to low-frequency word pairs" (Evert 2008).

[30] This dependency resembles the concept of lexical functions (Wanner 1996; Mel'čuk 1998), which models collocations as pairs of a word (denominated 'lexical unit') and the return value of a function applied to that word. Such a function would, in this case, return words like 'young', 'fresh', 'recent', 'juvenile', 'novel', 'nouveau', etc. depending on the argument it is applied to.

[31] That distance is commonly limited to either three or five tokens (Bartsch and Evert 2014).

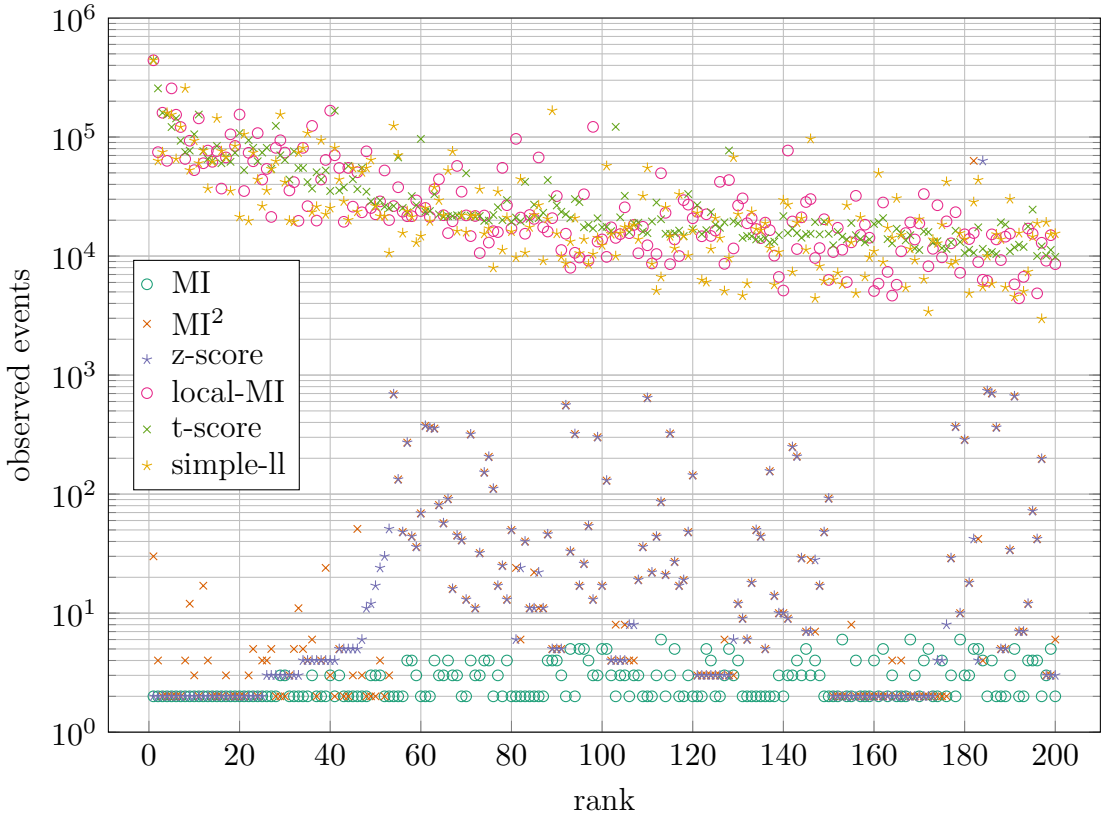


Figure 5.4 – Number of observed events (O) for 200-best list of six different simple association measures. The association measure can visually be divided into two groups: those with a preference for comparably low frequencies (MI, MI^2 and z-score) and those with a preference for high frequencies (local-MI, t-score, simple-ll).

same sentence. Table 5.6 shows two examples for pairs of words that occur together more frequently than expected ($O > E$). Inspecting the single example where 'rapaciously' and 'financing' occur together,[32] we see that they do not actually interact; which gives rise to the question if there can be a meaningful interpretation of textual cooccurrence on the sentence level. Cooccurrence-based word embedding techniques typically use token spans of a limited size and not whole sentences (Lebret and Collobert 2014; Li et al. 2015).

[32]"But if this Parliament is really concerned about the efficiency of the CFSP, and not only concerned with increasing its powers rapaciously, it should recommend a different attitude in order to overcome the contradiction between the institutional nature of the CFSP, and its method of financing."

Table 5.6 – Sample cooccurrence frequencies calculated on entire sentences. The expected cooccurrence frequencies are $E_{11} = 96.3$ for 'investment' and 'crisis', and $E_{11} = 0.002$ for 'rapaciously' and 'financing'. The former pair receives higher scores for all the six simple association measures previously used except for MI, which favors the latter.

	crisis	¬crisis
investment	221	17 150
¬investment	9070	1 649 366

	financing	¬financing
rapaciously	1	0
¬rapaciously	3395	1 672 411

The third way to measure cooccurrence detailed in (Evert 2008) in addition to surface and textual cooccurrence is to use syntactic relations, hence named *syntactic cooccurrence*. The difference to surface and textual cooccurrence is that each syntactic relationship is treated separately since they encode distinct relations. Applied to our corpus (FEP6), we find the same pattern as observed with surface cooccurrence: Some association measures show a preference for combinations of rare words and some prefer high frequencies. For adjectival modifiers, the highest MI score is attained by combinations of hapax legomena such as '46-year-old sergeant-major', 'virginal pedestal', ' rosy-cheeked milkmaid' or 'meditative ikebana'. Several of those examples are results of part-of-speech tagging and parsing errors (e.g., 'harlequin ladybird' and 'straw bedding'). It is therefore beneficial to apply a threshold for filtering out at least cases where $O = 1$ for syntactic cooccurrence.

Interlingual Association Measures

A large word-aligned parallel corpus lends itself to apply the same association measures to the company of strangers, to tie in with Firth's (1957) phrasing. This is basically a correlation that the word aligner learns (see Section 4.4), though typically not per se on single words but on phrases. However, the vast majority of the resulting alignment units are one-to-one alignments (depicted in Figure 4.21). We will refer to those frequencies of interlingual one-to-one alignments as *interlingual cooccurrence* to stress the point that they have been obtained from a particular corpus. Interlingual cooccurrence resembles syntactic cooccurrence insofar as it is based on relations between particular tokens. In contrast to syntactical relations, there are no distinguishable categories of alignment relations.[33]

Table 5.7 shows expected cooccurrence frequencies of the English word 'opponent' and two possible translations into German. The marginal frequencies, that is, the number of occurrences of a word that we count in the corpus, irrespective of

[33]Since cooccurrence describes a reciprocal property, we use symmetrized word alignments.

Table 5.7 – Contingency tables for two possible German translations of English 'opponent' based on interlingual cooccurrence. Expected frequencies are $E_{11} = 0.026$ for 'Gegner' and $E_{11} = 0.000\,74$ for 'Widersacher'. The respective association measures all rank 'Gegner' better, though MI scores are similar.

	opponent	¬opponent
Gegner	423	360
¬Gegner	290	21 183 754

	opponent	¬opponent
Widersacher	10	12
¬Widersacher	703	21 183 276

whether it cooccurs with some other word or not, are calculated on the subset of tokens that are aligned between these two languages. If we compare the resulting association scores with the frequent pair 'because'/'weil' ($O = 20\,830$, $E = 86$), we see the same characteristics as with surface cooccurrence: local-MI, t-score and simple-ll clearly favor the latter.

To achieve our objective of phraseme identification, we combine syntactic and interlingual cooccurrence. We use support verb constructions (SVC)[34] with direct objects to exemplify our approach (as illustrated first in Graën 2017). Support verbs in those constellations, which consist of a verb and its direct object, are verbs that make little or no contribution to the semantics of their sentences;[35] they are merely required to syntactically connect the predicative noun. The noun thus acts through the support verb and the verb syntactically realizes complements required by the noun. In the sentence "Profits take precedence over humanitarian concerns.", for example, the SVC 'to take precedence (over sth.)', which realizes the direct object, is accompanied by the subject ('profit') and a prepositional object ('over humanitarian concerns'). This object is a semantic argument of the noun ('precedence over …') and not the verb ('take … over …').

We exploit this discrepancy between syntax and semantics by making use of the fact that two corresponding (i.e., aligned) constellations of verb and direct object in parallel sentences, where at least one of them is an SVC, are likely to show dissimilar verb semantics.[36] Although we use a particular constellation of syntactic relations and word correspondences between two languages for demonstration purposes, this method is generally applicable to other parallel structures owing to the non-compositionality of phrasemes (Mel'čuk 1998).[37] Whenever one of two

[34] Frequently also referred to as light verb constructions.

[35] "The semantics of the support verb is either void or reduced to a small set of semantic features that are relevant for very large subclasses of verbs […]" (Langer 2004).

[36] This is not necessarily the case for all support verb constructions and each language pair. The English SVC 'to play a role', for instance, can be literally translated to SVCs in German ('eine Rolle spielen'), French ('jouer un rôle') or Polish ('odgrywać rolę').

[37] Melamed (1997a) concludes that "texts in two languages are not only preferable but necessary for discovery of non-compositional compounds for translation-related applications."

corresponding tokens in such a constellation only plays a functional (i.e., syntactic) role and it occurs in that role considerably less frequently than in free (i.e., semantic) use, we can automatically identify it as part of a potential phraseme.

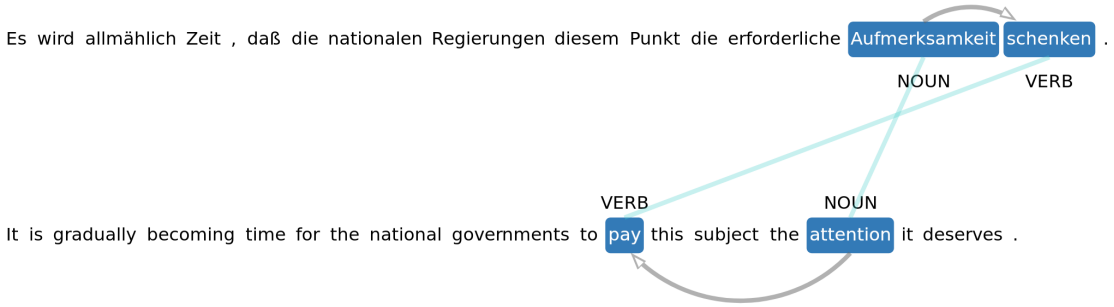


Figure 5.5 – Corpus sample for the corresponding phrasemes 'pay attention (to)' and 'Aufmerksamkeit schenken'. Direct object relationships in both languages and word alignments between both languages are indicated.

The SVC 'to pay attention (to)', for instance, is frequently translated as 'Aufmerksamkeit schenken' or 'Aufmerksamkeit widmen' (literally 'to give attention as a present' and 'to dedicate attention'). In this example, the lemmas 'attention' and 'Aufmerksamkeit' are standard translations and 'pay' and 'schenken' (or 'widmen') cannot be used as translations outside of a limited list of phrasemes ('Beachtung schenken' is an alternative, less frequent translation of 'to pay attention'). Figure 5.5 shows a corpus sample, which indicates how syntactic relations and word alignments together form a constellation. Here, we require a noun to be direct object of a verb in both languages and that both nouns and both verbs are aligned.

There are several approaches to extract multiword expressions (MEW) – to use the most general term – from parallel corpora. Melamed (1997a) employs IBM translation models (Brown, V. J. Della Pietra et al. 1993) to find sequences of words ('non-compositional compounds') that, treated as a single unit, increase the predictive power of a deduced translation model in terms of mutual information score. Villada Moirón and Tiedemann (2006) identify potential MWEs in the Dutch part of Europarl using statistical measures on relations obtained by syntactical dependency parsing. They use word alignment between Dutch and three other languages to calculate the translational entropy (Melamed 1997b) of each MWE as average of the individual words' entropy.[38] The calculated entropy is used to rank the previously identified MWE candidates. As assumed, a high

[38] "Translational entropy measures the predictability of the translation of an expression by looking at the links of its components to a target language." (Villada Moirón and Tiedemann 2006)

entropy is found to correlate with idiomaticity. Cap (2017) follows this approach to investigate compound compositionality of English and German nouns. To this end, she splits German compound nouns into their respective parts by means of morphological analysis, performs word alignment on the modified corpus and calculates the translational entropy with the German MWEs being the compound nouns split previously.

The objective of Zarrieß and Kuhn (2009) is to identify MWEs that translate to a single word in another language. Starting with a German verb, they calculate the alignment distribution into English and subsequently filter out unwanted constellations (based on, e.g., overly distant connections in the syntax tree). They found that their syntactic filters reliably support the identification of the MWE type in question. Medeiros Caseli et al. (2010) similarly use part-of-speech tag sequences to filter out MWE candidates. They deal, like (Melamed 1997a), with MWEs that consist of token sequences. Their approach relies on word alignments and sequences identified by the employed part-of-speech tagger, which recognizes, for instance, compound nouns. The method proposed by (Vargas et al. 2017) targets the identification of SVCs by means of part-of-speech patterns for the identification of potential MWEs in each language separately and a ranking of MWE candidate pairs based on cooccurrence and a multilingual word embedding model. This model "ensure[s] that words that are translations of each other end up being close in the resulting semantic space." (ibid.), which is counterproductive for the identification of SVCs from our point of view, since we expect the verbs of parallel SVCs to show non-standard translations, as we have detailed above. Accordingly, they report mixed, though "promising", results.

We search our corpus for such constellations using three language pairs: English/German, English/Italian and German/Italian. The most frequent constellations are shown in Table 5.8. We see that in many cases the respective verb pairs are frequent translations, which we also expect to find in other contexts (e.g., 'play'/'spielen', 'take'/'tenere', 'finden'/'trovare'), but also examples of non-standard correspondences such as 'treffen' *'to strike'*/*'to hit'* with 'prendere' *'to take'*. These are the ones that we want to use to identify phrasemes. To obtain a dictionary entry from those search results, that is, to get the entire phraseme, we also have to take into account other parts of the respective verb + noun pair such as fix determiners or prepositions (e.g., 'play a role', 'take into account', 'draw attention to', 'avere il diritto di').

In (Graën and Bless 2017), we present a web application (depicted in Figure 5.6) to explore association measures visually, based on both syntactic and interlingual cooccurrence frequencies.[39] The user can choose between English, German and Italian as source language, restrict the search to particular verbs and

[39] Available at http://pub.cl.uzh.ch/purl/visual_association_measures.

Table 5.8 – Most frequent constellations in FEP6 that involve direct object relationships and alignments between the respective verbs and nouns for all combinations of English, German and Italian.

English		German		Italian		
verb	dir. object	dir. object	verb	verb	dir. object	freq.
play	role	Rolle	spielen			2095
have	right	Recht	haben			1181
support	report	Bericht	unterstützen			1084
find	solution	Lösung	finden			983
support	proposal	Vorschlag	unterstützen			799
take	account			tenere	conto	2522
play	role			svolgere	ruolo	2432
thank	rapporteur			ringraziare	relatore	2142
have	right			avere	diritto	1890
draw	attention			richiamare	attenzione	1320
		Rolle	spielen	svolgere	ruolo	1726
		Recht	haben	avere	diritto	1241
		Problem	lösen	risolvere	problema	1088
		Lösung	finden	trovare	soluzione	845
		Entscheidung	treffen	prendere	decisione	716

nouns by means of regular expressions and change the ranking of the results. The standard ranking is performed by descending frequency. The other options are to apply one of five association measures, all of which are simple ones, based on O_{11} and E_{11}, in descending (standard) or ascending order. A threshold is provided to filter out infrequent constellations. We set the default threshold to two occurrences because a single occurrence may be the result of a random error in one of the statistically generated layers, although the circular constellation of alignments and dependency relations is still required. Two occurrences already require that error to be systematic.

Once a verb + noun pair from the source language has been selected, the distribution of aligned lemmas is shown for all available target languages, similar to the list of translation equivalents in Multilingwis (see Section 5.2). In addition to the source language ranking, which depends on either the source language syntactic cooccurrence frequency or one of both interlingual frequencies, the target language results can be ranked according to the same association measures. For the web application data, we remove the requirement of syntactic dependency between the

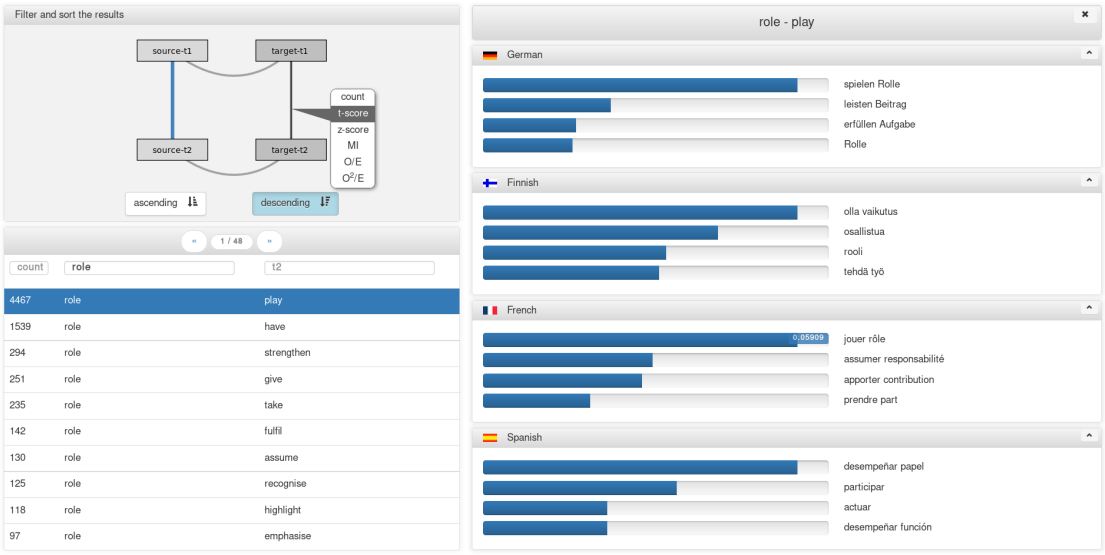


Figure 5.6 – Our web application to explore the effect of different association measures on both syntactic and interlingual cooccurrence frequencies.

aligned tokens; we combine all aligned tokens to a list instead (again similar to what Multilingwis does). That way, we get noisier, but at the same time more data. We find, for instance, '(to) bear witness' with the single word translations 'zeugen', 'bezeugen', 'zeigen' in German and 'testimoniare', 'dimostrare' in Italian, which we possibly would have missed with the syntactic restriction in place. At the same time, we also get lists with more than two words, for instance, when the German source language noun is a compound: 'Todesstrafe abschaffen' is aligned with 'abolish death penalty' and 'abolish capital punishment'.

We designed the application to provide a means for exploring the preferences of different association measures and the difference between syntactic and interlingual cooccurrence. In particular, we are interested in finding the right association measure that disqualifies best those verbs that serve as translations outside their phrasemic use. For the language pair English/German, we want, for instance, to disregard 'geben' as predominant translation of 'give', so that 'give work' and 'Arbeit geben' will receive a low score, while 'setzen' *'set'*/*'put'* as translation of 'give' will result in higher scores for combinations (e.g., 'Zeichen setzen' *'to give sign'*, 'Signal setzen' *'to give Signal'* or 'Prioritäten setzen' *'to give priority'*).

What we are looking for should, hence, rather be named a *dissociation measure*. For that purpose, we decided to combine different association measures. The challenge in doing so is that they calculate a score that is meaningful in comparison to other scores calculated on the same cooccurrence frequencies (i.e., the same corpus), but meaningless in comparison to scores of other association measures

Table 5.9 – Highest ranked verb + noun pairs for all combinations of English, German and Italian using a joint score of syntactic and interlingual association measures.

English		German		Italian		
verb	dir. object	dir. object	verb	verb	dir. object	freq.
take	shape	Gestalt	annehmen *'adopt'*			39
set	precedent	Präzedenzfall	darstellen *'represent'*			10
reduce	poverty	Armut	bekämpfen *'combat'*			4
set	precedent	Präzedenzfall	schaffen *'create'*			78
take	precedence	Vorrang	haben *'have'*			47
take	look			dare *'give'*	occhiata	21
take	precedence			dare *'give'*	precedenza	4
send	condolence			esprimere *'express'*	condoglianza	5
take	precedence			avere *'have'*	precedenza	92
have	illusion			fare *'make'*	illusione	20
		Abhilfe	schaffen *'create'*	porre *'put'*	rimedio	36
		Präzedenzfall	schaffen *'create'*	costituire *'establish'*	precedente	23
		Oberhand	gewinnen *'win'*	prendere *'take'*	sopravvento	8
		Mühe	machen *'make'*	prendere *'take'*	briga	9
		Klarheit	schaffen *'create'*	fare *'make'*	chiarezza	6

or corpora.[40] To solve this problem, we use the respective association measures for ranking and convert them to values between zero and one using a cumulative percentile ranking (i.e., a linear mapping of the respective ranks).

For our constellation, we found, setting the association score of the nouns (N) in relation to the score of the verbs (V) a good starting point. To restrict the values that the ratio of these two scores can take, we add a constant value δ to both. We multiply this fraction with the weighted sum of normalized association scores for source (S) and target language (T) pairs and get a combined score for the constellation:

$$score = (w_S \cdot m_s(S) + w_T \cdot m_s(T)) \cdot \frac{\delta + m_i(N)}{\delta + m_i(V)} \tag{5.5}$$

For the combined score, we allow different association measures for syntactic (m_s) on the one hand and interlingual cooccurrence frequencies (m_i) on the other hand, but not for the respective pairs on the same dimension. In Table 5.9, we list the best scoring constellations with local-MI as m_s, O/E as m_i,[41] and both weights and δ set to 1. Our impression from larger n-best lists than the ones presented in Table 5.9 is that we find candidates with good prospects for a dictionary entry, even those that show a low overall frequency such as '(to) reduce poverty' and the (metaphorical) phraseme 'Armut bekämpfen' in German. The standard (i.e., literal) translations of 'reduce', 'reduzieren' and 'verringern', do also occur in our corpus, but they are less phrasemic and their meaning can be inferred from the meaning of their constituents.

We also observe that, similar to the case of 'to reduce poverty', we see cases where the expression in one language is more phrasemic than in the other one. This is owed to the method, which only requests that the verbs do not correlate well. If we were to say which of the two expressions to propose as a dictionary entry, we would need to consult both syntactic cooccurrence frequencies and the marginal frequencies of the respective verbs.

Outlook

In (Graën 2017; Graën and Bless 2017), we have presented a novel approach to identify phrasemes by means of interlingual association scores, which are common association scores applied to word-alignment-based cooccurrence frequencies. Word alignment on parallel corpora has proven useful for this objective. By com-

[40] An exception may be to a certain extent the association measures based on information theoretic considerations (Evert 2008).

[41] Which gives the same ranking as MI since the logarithm function is continuous.

bining syntactic and interlingual association measures,[42] we found a promising method to identify support verb constructions as a special case of phrasemic structures. We believe that the method will generalize to phrasemes of an arbitrary syntactic structure. In cases where two languages agree on the choice of words (e.g., by reason of language history), an examination of several language pairs will prove helpful.

A systematic evaluation of the resulting n-best lists is necessary to assess the method's success rate. To this end, different combinations of association measures and weights should be compared to gold standard phraseme lists in the respective languages (Bartsch and Evert 2014). Nonetheless, a clear definition of which expression should be included in a dictionary and which one is self-explanatory to a language user can arguably not be found as semantic transparency needs to be thought as a continuum. Idiomaticity is typically not as obvious as in the case of support verb constructions.

As our main interest is the identification of monolingual phrasemes, we will also need to test syntactic cooccurrence frequencies and the marginal frequencies of the respective verbs as indicators for which of the two aligned verb + noun combinations is phrasemic and which one is not. We also consider combining the results of retrieval on several language pairs to distinguish between good candidates and combinations that are only useful in contrast with them. To get actual dictionary entries from our verb + noun combinations (e.g., including determiners and prepositions), a frequency analysis on syntactic cooccurrences needs to be performed for each combination that has been identified.

5.4 Backtranslating Prepositions for Prediction of Language Learners' Transfer Errors

Multiword expressions are among the lexical items that language teaching materials such as textbooks introduce to learners. They are typically chosen to enhance the learner's expressivity gradually. 'False friends', that is, word or multiword expressions that look similar to the learner's native language (L1) but convey a different meaning (see also Section 5.1), may be addressed if the teaching material has been created with that particular L1 in mind. While a limited number of false friends can be listed so that the learner memorizes them, the correct use of function words, notably prepositions, is comparably harder to accomplish as many prepositions are predetermined by other words, in particular verbs and adjectives.

[42] Using surface cooccurrence instead of syntactic cooccurrence will presumably generate results similar to the phrase tables that bilingual word aligners learn from sentence-aligned corpora (see Section 4.4). Bartsch and Evert (2014) also state that "assuming a syntactic dependency context is optimal for an identification of Firthian collocations."

Preposition usage can be divided into two classes: those that are semantically transparent (e.g., 'on' in 'sit on') and those that are intransparent (e.g., 'for' in 'wait for').[43] Phrasal verbs (e.g., 'depend on') cannot be decomposed semantically into verb meaning and preposition meaning either. As we see in these examples, both classes overlap; the preposition 'on' can be a spatial reference (e.g., 'on a table') or merely a linguistic necessity as in 'depend on'.[44] Prepositions show vast cross-lingual variation and are "notoriously difficult to master" for non-native speakers (Swanepoel 1998). Furthermore, "[they] are often considered [...] impossible to teach and impossible to learn" (Gilquin and Granger 2011).

One frequent source of lexico-grammatical errors are combinations of verbs or adjectives with prepositions, henceforth referred to as VPC and APC, respectively. We understand these combinations in the broader sense of the term, including the notions of compound verbs and phrasal verbs. According to Gardner and Davies (2007), phrasal verbs represent "one of the most notoriously challenging aspects of English language instruction". In what follows, we describe our approach to predicting transfer errors in VPC/APC committed by learners with different L1 backgrounds. The prediction is entirely based on our corpus (FEP6), we access learner corpora only for evaluation. For the sake of clarity, we will explain our approach (as presented in Graën and Schneider 2017) using the example of VPC. The treatment of APC is carried out accordingly.

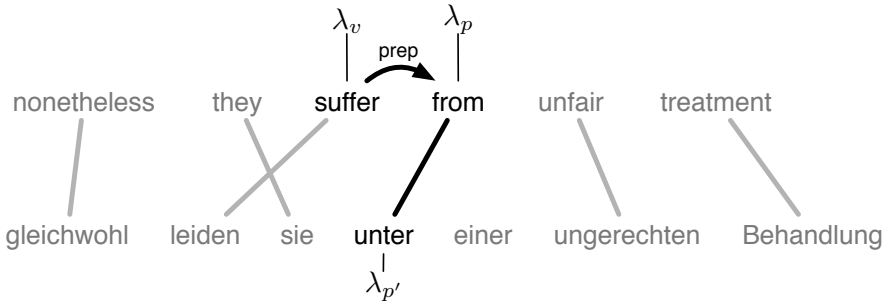


Figure 5.7 – Sample constellation from our corpus. The dependency between verb and preposition and the alignment between both prepositions are marked.

We first identify VPCs by searching for prepositions, so-called prepositional modifiers in the Stanford typed dependencies schema (Marneffe and Manning 2008), of a verb in the English part of our corpus. Using word alignment information (see Section 4.4), we retrieve the corresponding prepositions in all other

[43] Gilquin and Granger (2011) identify eight semantic classes for the preposition 'into'; Swanepoel (1998) reports 35 different senses of 'in' in *The Concise Oxford Dictionary*, 41 senses in the *Longman Dictionary of Contemporary English* and 56 senses in the *Collins Cobuild English Language Dictionary*.

[44] Pinker (1996) reasons why children acquire semantically transparent prepositions earlier.

languages. For a constellation to qualify as result, all these conditions must be met. Figure 5.7 depicts such a constellation. We use the symbols λ_v, λ_p and $\lambda_{p'}$ to refer to the English verb, English preposition and the aligned foreign language preposition, respectively.

The verb + preposition pairs obtained this way provide us with corpus frequencies of different prepositions per verb. We also calculate corpus frequencies for the raw occurrences of the respective verbs. That way, we can calculate the ratio of verb usage with and without prepositions. The verb 'to suffer', for instance, occurs in 26 % of its occurrences with the preposition 'from'. Other less frequent prepositions are 'in' (9 %) and 'under' (2 %). The preposition 'from' shows the highest frequency, and we therefore assume that it is the correct one, if 'suffer' is used in combination with a preposition, and continue working with that VPC.

Table 5.10 – Backtranslation score (BTS) and backtranslation ratio (BTR) for different backtranslated prepositions $\lambda_{p''}$ of 'to suffer from'. The list on the left is derived from English/German and the one on the right from English/French alignments.

λ_v	λ_p	$\lambda_{p''}$	BTS	BTR
suffer	from	under	102.51	2.51
suffer	from	of	100.04	2.46
suffer	from	in	78.56	1.93
suffer	from	by	51.19	1.25
suffer	from	on	46.53	1.14
suffer	from	from	40.97	1.00
suffer	from	with	36.32	0.89
suffer	from	among	27.93	0.68

λ_v	λ_p	$\lambda_{p''}$	BTS	BTR
suffer	from	of	457.31	13.11
suffer	from	for	68.79	1.97
suffer	from	by	58.44	1.68
suffer	from	in	57.56	1.65
suffer	from	from	34.88	1.00
suffer	from	on	25.58	0.73
suffer	from	with	18.24	0.52
suffer	from	about	7.23	0.68

Having identified the correct verb preposition combination, we derive the distribution of aligned target language prepositions for each target language (here: French, German, Italian, Polish and Spanish). We subsequently multiply these distributions with the lemma alignment distribution matrix (see Section 3.2.1) and obtain lists of English prepositions with weights in each language. We call them backtranslation scores (BTS) since we were starting from English preposition at the outset.

A backtranslation score has no meaningful value (absolute frequency of syntactic occurrences multiplied with lemma alignment probability) but allows us to rank the resulting prepositions. We normalize the values by comparing each score with the score achieved by the correct preposition, that is by calculating the ratio of both scores, and obtain the backtranslation ratio (BTR), which defaults to 1

for the correct preposition. We interpret values higher than 1 as likelihood of the corresponding preposition to be confused by a language learner whose native language is the language in question. The highest scoring preposition is the one we expect to be mistakenly used most frequently.

Table 5.10 lists backtranslation scores and ratios for the VPC 'suffer from' and German and French as intermediate language. The most likely English preposition to be mistakenly used by German learners of English is 'under', which is presumably due to the frequent translation 'leiden unter' of 'suffer from' (see also Table 5.10). The German preposition 'unter' is predominantly used as a spatial reference like its English counterpart 'under' and rarely in phrasal verbs. While in some cases the most salient wrong preposition predicted by our algorithm can easily be traced back to a single verbal expression in the respective language, this is not true for the majority of VPCs. The predicted wrong English preposition is the result of all occurring translations together with the general translation preferences of the prepositions in other languages.

We generate lists of VPCs and APCs ranked by a combination of the respective verb's corpus frequency and the highest BTR for that VPC or APC. As these lists are meant as recommendations for language learners, we limit it to approximately 100 VPCs and 25 APCs per language. Sample recommendation lists are provided in Appendix C.2.

Evaluation

We evaluate our approach in two ways: On the one hand, we look up predicted error-prone VPC and APC from our lists in learner corpora to verify that these predict errors that language learners actually commit. On the other hand, we propose corrections to a list of incorrect combinations (Schneider and Gilquin 2016) based on the respective preposition that our algorithm assumes to be the correct one and judge them manually.

The basis for our evaluation of actual learner errors are three learner corpora:[45] the *International Corpus of Learner English (ICLE)* (Granger et al. 2002), the *First Certificate in English (FCE)* dataset (Yannakoudakis et al. 2011) and the *NICT Japanese Learner English (JLE) Corpus* (Japanese National Institute of Information and Communications Technology 2012). Since we do not have any resource available that would provide us with a sufficiently large number of learner errors subdivided by language, we derive a language-independent list of VPC and APC that appear in each of the language-specific lists. The number of VPC in the intersected list amounts to 40 and the number of APC to seven.

[45] Those are typically error-annotated collections of text or transcribed speech produced by language learners, which are studied for understanding the properties of language learning.

Table 5.11 – Language-independent verb preposition and adjective preposition combinations present in all five language-specific recommendation lists. 23 out of 31 relevant ones can be found in at least one of the learner corpora we searched (I: ICLE; N: NICT JLE; F: FCE). Entries that we consider valid are checked, questionable ones receive a question mark.

VPC/APC		OK?	I	N	F
aim	at	✓	✓		
arrive	at	✓	✓	✓	✓
benefit	from	✓	✓		
breathe	into	?	*n/a*		
channel	into	✓	*n/a*		
complain	about	✓	✓	✓	✓
compliment	on	✓			
convert	into	✓	*n/a*		
depend	on	✓	✓		✓
direct	at	✓	✓		
divide	into	?	*n/a*		
emanate	from	✓			
embark	on	✓			
enter	into	?	*n/a*		
estimate	at	✓	✓		
exclude	from	✓	✓		
exempt	from	✓	✓		
fall	within	✓			
force	into	✓	*n/a*		
gain	from	✓	✓		
hang	over	✗	*n/a*		
incorporate	into	?	*n/a*		
integrate	into	?	*n/a*		
level	at	✗	*n/a*		

VPC/APC		OK?	I	N	F
look	at	✓	✓	✓	✓
miss	from	✓			
plunge	into	?	*n/a*		
preside	over	✓			
profit	from	✓	✓		
protect	from	✓			
recover	from	✓			
suffer	from	✓			✓
talk	about	✓	✓	✓	✓
target	at	✓	✓		
throw	into	?	*n/a*		
transform	into	?	*n/a*		
translate	into	?	*n/a*		
transpose	into	?	*n/a*		
wait	for	✓	✓	✓	✓
worry	about	✓			✓
absent	from	✓	✓		
conditional	on	✓	✓		
dependent	on	✓	✓	✓	✓
early	as	✗	*n/a*		
exempt	from	✓	✓		
sceptical	about	✓	✓		
serious	about	✓	✓		

We judge every entry's validity. An entry is considered valid if it contains a non-semantic, non-compositional preposition or if the preposition is language-specific. We disregard, for instance, the VPC 'level at' as the verbal expression 'to level at somebody' is exceptionally frequent in our corpus,[46] but we do not believe it is in typical language use.[47] In this regard, the parliamentary register may have an impact that renders particular results of our approach less suitable for language learners. We were particularly unsure about the status of the English preposition 'into', which does not exist as preposition in most other languages, but which is at the same time typically semantically transparent. Hence, we do not use VPC with 'into' in our evaluation.[48]

Subsequent to our judgment, we look up the remaining relevant entries, that is, the ones we consider valid except for those that use the preposition 'into', in all three learner corpora. If we find an occurrence where a learner commits an error regarding the preposition of the verb or adjective in question, we count that as evidence for the difficulty of that VPC or APC. Table 5.11 shows our intersection list of VPCs and APCs together with our judgment and the results from consulting the respective learner corpora. We observe that frequent combinations can be found in all three corpora, while the less frequent – but highly error-prone – ones are most frequently found in ICLE. In total, 23 out of 31 relevant VPCs and APCs were at least once incorrectly used in at least one of the three corpora, which results in a precision of 0.74.

Our second evaluation, the automatic correction of learner errors, uses a data set of incorrect VPC and APC compiled by (Schneider and Gilquin 2016). The data set comprises 48 English verbs and adjectives with erroneous prepositions as produced by language learners. For each verb and adjective from that list, our correction strategy is to simply propose the preposition that we previously have identified as correct or, in case the verb or adjective is used predominantly without a preposition according to our corpus statistics, not to use any preposition.

Out of the 48 entries with erroneous preposition, our approach proposes the correct preposition in 38 cases, which is depicted in Table 5.12. We partly attribute the remaining errors to the parliamentary register. The manual correction for the verb 'call' is not to use any preposition. Our approach, however, suggests 'call for' as the correct VPC. In our corpus, more than 40 % of the occurrences of 'call' are followed by the preposition 'for'. In the case of 'bad for', our approach is misled by

[46] As in "This criticism can no longer be levelled at us." or "There is no room for self-righteousness in the criticisms we level at Ukraine today, for we, thank God, have been spared a fate such as theirs - or, at least, most of us have been."

[47] It might also be the case that our frequencies are distorted and the plural noun 'levels' is occasionally confounded by the part-of-speech tagger with the third person singular verb, which would have a strong impact on distributional statistics for the infrequent verb 'level'.

[48] The only APC with 'into' we identified in our corpus is 'deep into'.

Table 5.12 – Incorrect VPC and APC, originally from (Schneider and Gilquin 2016), together with their manual correction in the second column. OBJ means that the manual correction of learner errors suggests using a direct object instead. Cases marked with *n/a* do not fall into the preposition correction scheme as the manual correction affects the verb or adjective rather than the preposition. An entry is checked if our automatic correction matches the manual one, which is the case in 38 out of 48 VPC and APC (precision is consequently 0.79).

Incorrect VPC/APC		Correct		Incorrect VPC/APC		Correct	
accuse	for	of	✓	interest	for	in	
addict	on	to	✓	involve	into	in	✓
alarm	of	at	✓	relate	with	to	✓
apply	into	to	✓	replace	to	by	
assist	to	OBJ	✓	resist	to	OBJ	✓
assure	to	OBJ	✓	select	among	from	
aspire	for	to	✓	separate	between	*n/a*	
attack	against	OBJ	✓	study	about	OBJ	✓
aware	about	of	✓	understand	towards	OBJ	✓
belong	into	to	✓	view	upon	on	
benefit	out	from	✓	bad	to	for	
call	like	OBJ		capable	in	of	✓
characterize	with	by	✓	conscious	about	of	✓
charge	of	with	✓	critical	against	of	✓
confront	to	with	✓	critical	towards	of	✓
consist	on	of	✓	dependent	from	on	✓
deal	about	with	✓	dependent	of	on	✓
deprive	from	of	✓	diverse	by	*n/a*	
destructive	for	to	✓	guilty	for	of	✓
discuss	about	OBJ	✓	independent	on	of	✓
estimate	to	at	✓	responsible	of	for	✓
extend	of	to		superior	than	to	✓
impose	to	on	✓	synonymous	to	with	✓
indulge	into	in	✓	worth	for	OBJ	

the frequent expression 'worse than', which is lemmatized to 'bad' and 'than' and tagged as adjective and preposition. The Penn Treebank tag assigned to 'than' is 'IN', which is used for both preposition and subordinating conjunctions such as 'than'. This is why the parsing model establishes a 'prepositional modifier' relation between 'than' and 'bad'. In fact, all adjectives with 'than' tagged as preposition originate from comparatives. If we exclude 'than' as valid preposition for APCs, our correction strategy choses 'for' for the adjective 'bad', which is the correct preposition in this case.

Chapter 6

Conclusions

The main interest of this thesis was to investigate different aspects of alignment in multiparallel corpora. This chapter gives a summary of the previous chapters and provides answers to the research questions we defined in Chapter 1.

Corpus Preparation

We built a corpus with text, sentence and word alignment in 16 languages.[1] The preparatory steps include the correction of errors in the original corpus (Europarl),[2] tokenization, sentence segmentation, part-of-speech tagging, lemmatization, syntactic dependency parsing and alignment on different structural levels.

Tokenization is often regarded as a solved problem and not paid the attention it deserves from our point of view, as we frequently find subsequent processing errors due to erroneous tokenization decisions. Token types (i.e., different identifiable kinds of tokens such as numbers, abbreviations or punctuation marks) can be divided into language-specific and language-independent ones. Language-independent token types also comprise the class of corpus-specific tokens, for instance, domain-specific identifiers, which we only find in a particular corpus or in corpora of a particular domain. In the debates of the European Parliament, these identifiers are references to documents or resolutions. For these reasons, we have built our own pattern-based (i.e., rule-based) tokenizer, Cutter.[3] Cutter, unlike statistical tokenizers that have learned how to tokenize from a particular manually tokenized resource, can easily be adapted to new text types, including new languages.

[1] The latest corpus version will be made available together with our sentence and word alignment gold standards in different formats through the project website: http://pub.cl.uzh.ch/purl/sparcling_project

[2] The corrected Europarl corpus is available at http://pub.cl.uzh.ch/purl/costep.

[3] Cutter can be obtained from http://pub.cl.uzh.ch/purl/cutter.

The TreeTagger, as joint part-of-speech tagging and lemmatization tool, assigns multiple lemmas to word forms that are ambiguous with regard to the determined tag. We adapt an existing approach to resolve these lemma ambiguities by means of bilingual word alignment in parallel corpora and extend it to multiparallel corpora. For this disambiguation approach, we sum up the conditional probabilities of all aligned words disregarding the respective language. An interesting question for future work is how decisive a particular language is for disambiguation, for instance, if Spanish is more decisive for disambiguation of German lemma variants than Slovene, and if an accordingly weighted sum would yield even better results.

Use Cases of Word Alignment in Multiparallel Corpora

The study of language use in corpora is nowadays not limited to corpus linguists anymore. The SketchEngine as commercial corpus exploration tool, for instance, lists a variety of potential user groups, among them expert and non-expert users. We present several applications in this thesis that target both types of users (as detailed below). In particular, we focus on linguists and language learners.

By intersecting the alignment distributions of two words, which we obtain from bilingual word alignment on all available language pairs, we measure the similarity of meaning in terms of translation preference that these words have in common. This method also works for words of the same language, which gives us a measure of synonymy or interchangeability.

This so-called alignment distribution overlap yields promising results, though thorough testing needs still to be done in the future. We were able to attest, for instance, that some expected false friends actually have little to no overlap. Other false friends candidates showed an overlap that was larger than expected. We subsequently learned about contexts in which these false friends candidates are valid translation of each other by inspecting the corpus examples where those pairs are aligned. As future work, we propose to use this method to identify false friends in learner texts, for instance, by testing the standard translation of a false friend in place of the original word and collocation measures applied to both alternatives. We also suggest compiling actual false friends lists with examples showing the different uses of each word, which can be used as didactic material, provided good results in evaluation.

Our method for predicting learner errors with regard to preposition use in combination with verbs and adjectives likewise addresses transfer errors of language learners. We use the alignment distribution in combination with syntactic dependency parsing to rate distributions of prepositions in relation to the preposition we automatically identified as the correct one. We interpret the best-rated preposition as the most probable error and show, by comparison with examples from learner corpora, that the errors we predict are actually made by language learners

and that our method is able to correct learner errors in most cases. In addition, we compiled lists of verb preposition and adjective preposition combinations for language learners of English and five different native languages. These lists comprise those combinations that we expect to be similarly error-prone and important in terms of frequency, as estimated by our model.[4]

The method we proposed for identification of phrasemes follows a similar approach, insofar as it combines two dimensions of annotation: syntactic dependency parsing and word alignment. The target user group, however, are in this case rather corpus linguists than language learners, since our approach deals with the combination of statistical association measures applied to both dimensions. As test case, we use support verb constructions to benefit from the semantic nonconformance of their verbs between languages.[5] We tried a particular combination of association measures that yields no false positive result among the top-rated results. An in-depth evaluation, however, requires a standard that defines which combinations are phrasemic and which are not, a question that cannot be determined easily. Moreover, it needs to be investigated how well this approach performs on other syntactic constructions.

With our corpus exploration tool Multilingwis, we provide a means to search and explore multiword expressions and their translations in several languages of a multiparallel corpus. It has been designed with several user groups in mind, but has proven to be particularly useful for language learners and linguists. It provides, for example, answers to the questions how an expression is translated into other languages or how it is typically used in context. Translation variants and their frequencies are listed and allow for a faceted search. Future work needs to address the demands of different user groups with respect to a more powerful query language. Beyond that, parallel queries on multiple corpora will facilitate comparison between translation variants among corpora.

In all our work, we have used relational databases to represent the corpus structure, hold corpus data and cached query results, and perform efficient queries over multiple layers of annotation and alignment. In particular, Multilingwis is driven by the sophisticated indexing techniques that our database management system of choice, PostgreSQL, offers. This is also the only essential technological requirement to deploy Multilingwis on other sites.[6]

[4]The verb preposition and adjective preposition combinations together with their respective frequencies are listed at: http://pub.cl.uzh.ch/purl/reimporting_prepositions

[5]Different association measures on all possible connections between each two tokens can be tested through a web interface: http://pub.cl.uzh.ch/purl/visual_association_measures

[6]Multilingwis is made available at https://pub.cl.uzh.ch/purl/multilingwis.

Multilingual Alignment

In this thesis, we presented approaches to multilingual alignment on different levels. While the alignment of speaker contributions merely means correcting the implicit alignment information given in the original Europarl corpus, which we did primarily by means of fuzzy matching the speaker names, the multilingual alignment on sentence and word level requires novel approaches; sentence and word alignment on multiple languages simultaneously poses an additional challenge.

We present an approach to multilingual sentence alignment, which performs hierarchical agglomerative clustering using the single-linkage method on a weighted sum of several features. To evaluate its performance, we manually aligned[7] 100 parallel texts in 16 languages as gold standard and subsequently calculated average F-Scores for the multilingual alignments found by our algorithm and their projection to language pairs (i.e., bilingual alignments). As a reference, we used a bilingual sentence aligner, hunalign, on the same data. The results are similar (our approach performs better but with marginal differences), though the task of aligning 16 languages consistently is arguably harder than aligning only two.

We also present an approach to multilingual word alignment likewise using hierarchical agglomerative clustering, but here we apply a variant of average-linkage clustering. This method is more elaborate than single-linkage clustering as it uses a distance measure between clusters instead of their respective components, which is why it requires recalculation of distances after every merge. On the other hand, it allows us to define more fine-grained constraints on the clustering process. Taking the gold standard for multilingual sentence alignment as a basis, we manually aligned tokens in 500 of those parallel sentences in six languages to obtain a multilingual word alignment gold standard. Single words often correspond to single words in other languages. However, alignments different from one-to-one are also frequent and, additionally, alignments can be part of other alignments with a broader scope. We therefore allowed arbitrary nesting of word alignments in our gold standard.

For evaluation, we compared four different bilingual word aligners and our multilingual word alignment approach to bilingual projections of the gold standard. None of the bilingual word aligners performs well at the identification of our gold alignment units, but by means of a second evaluation on single alignment links, we see that they are frequently partially matching. Our multilingual word alignment approach performs worse on bilingual alignment than the bilingual aligners, but is able to identify two third of the multilingual gold alignment units correctly (with many false positives). Applying the envisaged but not yet implemented filtering of intermediate clustering results, which we expect to reduce the number

[7] We provide a portable version of our tool for multilingual alignment of arbitrary units at http://pub.cl.uzh.ch/purl/hierarchical_alignment_tool.

of false positives considerably, remains for future work. The obtained results are insofar promising as the task of word alignment is less well-defined than other levels of alignment and the number of ways to do it wrong is exceedingly high for multilingual word alignment. In addition, the analysis of single alignment links from our calculated multilingual word alignments suggests that often only single errors are responsible for the disqualification of the whole alignment units.

Appendices

Appendix A

Linguistic Annotation

A.1 Universal Dependency Labels Produced by our Parsers for French, Italian and Spanish

The following table lists the label sets of syntactic relations used for parsing French, Italian and Spanish in FEP9. Universal dependency (UD) relations defined by version 1 and 2 of the standard (Marneffe, Dozat et al. 2014) that never appear in our parsed texts are shown as well. The parsing pipelines are described in Section 3.3 and – in more detail – in (Baffelli 2016). Language-specific dependency relations are subtypes of universal dependency relations and indicated by italic labels.[1] Universal dependency relations are documented at http://universaldependencies.org.

[1]The label 'ROOT' is presumably a relict of rare cases of wrong annotation in the training corpus. In total, we only find 74 relations with that label in our corpus out of which 72 have a noun as their head.

	FEP9			UD		
Label	French	Italian	Spanish	v1	v2	Description
acl	✓	✓	✓	✓	✓	clausal modifier of noun
acl:relcl	✓	✓	✓			
advcl	✓	✓	✓	✓	✓	adverbial clause modifier
advmod	✓	✓	✓	✓	✓	adverbial modifier
amod	✓	✓	✓	✓	✓	adjectival modifier
appos	✓	✓	✓	✓	✓	appositional modifier
aux	✓	✓	✓	✓	✓	auxiliary
auxpass	✓	✓	✓	✓		passive auxiliary
auxpass:reflex			✓			
case	✓	✓	✓	✓	✓	case marking
cc	✓	✓	✓	✓	✓	coordinating conjunction
ccomp	✓	✓	✓	✓	✓	clausal complement
clf					✓	classifier
compound	✓	✓	✓	✓	✓	compound
conj	✓	✓	✓	✓	✓	conjunct
cop	✓	✓	✓	✓	✓	copula
csubj	✓	✓	✓	✓	✓	clausal subject
csubjpass		✓	✓	✓		clausal passive subject
dep	✓	✓	✓	✓	✓	unspecified dependency
det	✓	✓	✓	✓	✓	determiner
det:poss		✓				
det:predet		✓				
discourse	✓	✓		✓	✓	discourse element
dislocated				✓	✓	dislocated elements
dobj	✓	✓	✓	✓		direct object
expl	✓	✓		✓	✓	expletive
expl:impers		✓				
fixed					✓	fixed multiword expression
flat					✓	flat multiword expression
foreign		✓		✓		foreign word
goeswith	✓			✓	✓	goes with
iobj	✓	✓	✓	✓	✓	indirect object

	FEP9			UD		
Label	French	Italian	Spanish	v1	v2	Description
list				✓	✓	list
mark	✓	✓	✓	✓	✓	marker
mwe	✓	✓	✓	✓		multiword expression
name	✓	✓	✓	✓		name
neg	✓	✓	✓	✓		negation modifier
nmod	✓	✓	✓	✓	✓	nominal modifier
nmod:poss	✓					
nsubj	✓	✓	✓	✓	✓	nominal subject
nsubjpass	✓	✓	✓	✓		passive nominal subject
nummod	✓	✓	✓	✓	✓	numeric modifier
obj					✓	object
obl					✓	oblique nominal
orphan					✓	orphan
parataxis	✓	✓	✓	✓	✓	parataxis
punct	✓	✓	✓	✓	✓	punctuation
remnant				✓		remnant in ellipsis
reparandum	✓	✓	✓	✓	✓	overridden disfluency
root	✓	✓	✓	✓	✓	root
ROOT		✓				
vocative	✓	✓		✓	✓	vocative
xcomp	✓	✓	✓	✓	✓	open clausal complement

A.2 Our Hierarchical Alignment Tool

The screenshot shows how the graphical user interface of our tool for manual multilingual alignment looks like. It consists (from top to bottom) of a menu bar, the item list where each token of the respective sentences is a selectable item, an information bar and the alignment area. The alignment inspector (not shown), opens in a separate window.

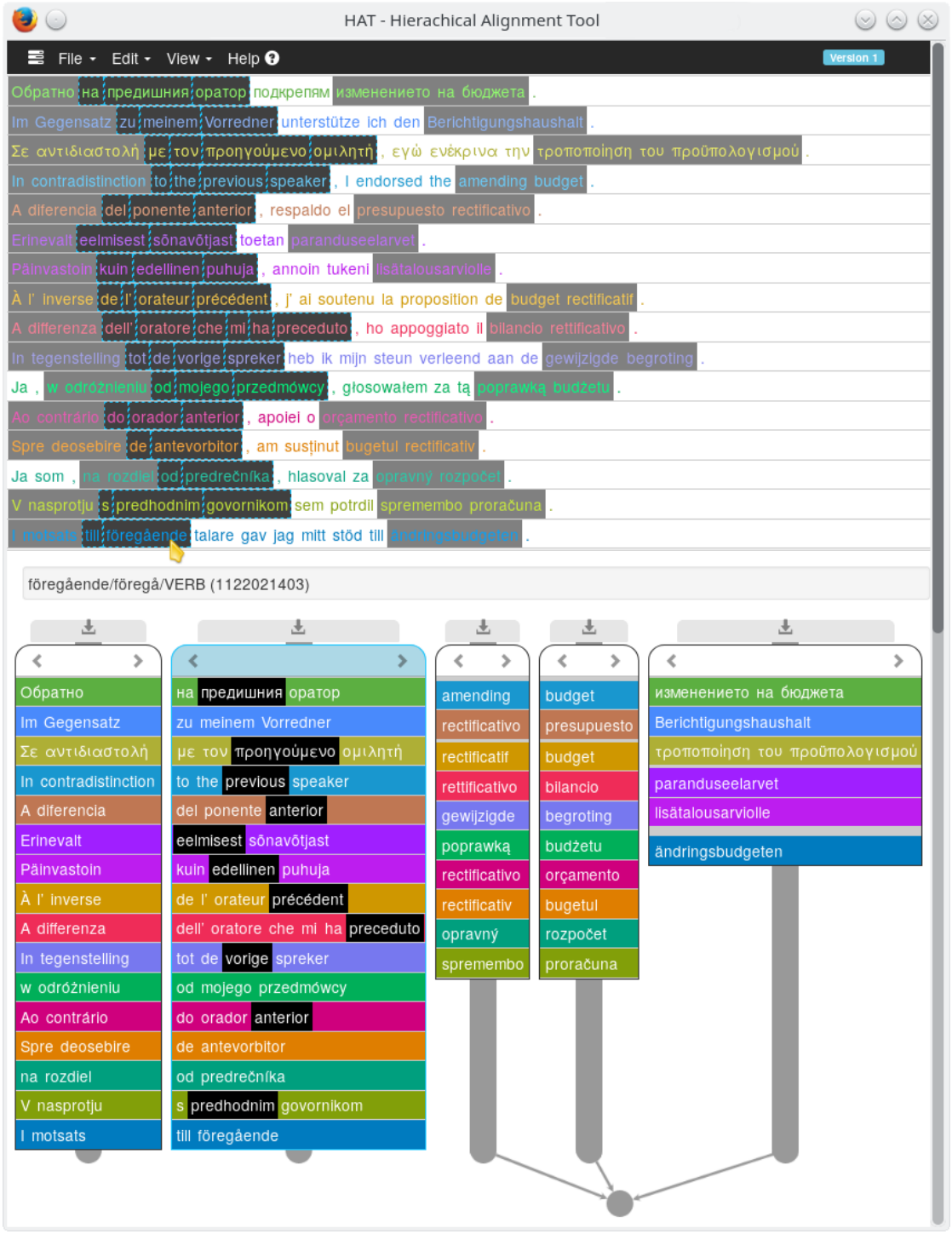
The Hierarchical Alignment Tool (HAT) is designed for large resolutions and supports a twin screen workstation layout so that the alignment inspector, which shows the selected tokens of both the item list and the selected AUs in the alignment area, can be opened on the second screen. In this way, it facilitates the annotator's decision whether the selected set of tokens bears the same context-independent meaning or not.

As shown, HAT supports multilingual word alignment in 16 languages (and possibly more), although we limited ourselves to the alignment of six languages for our set of gold alignments. The screenshot depicts a partial alignment of 16 parallel sentences. The annotator is in the middle of separating corresponding tokens ('previous') of a larger AU ('to the previous speaker') into a new, subordinate AU. Token-specific information (word form, lemma, part-of-speech and the numeric identifier of the token) of the respective token under the cursor is displayed in the middle bar.

Although this example demonstrates its ability to align words, the same tool can be used for multilingual sentences alignment. In that case, sentences are cropped after an adjustable length and the lengths of all sentences can be indicated on request. Suggestions have been made that it could also serve for aligning errors in learner texts.

HAT uses a simple JSON-based interface for retrieving the initial list of items in all languages, for saving multilingual ASs and for fetching a possibly existing AS. Multiple serially numbered versions of a particular AS are supported, thus allowing for aligning previously unaligned languages at a later date. The source code is available online.[2]

[2] http://pub.cl.uzh.ch/purl/hierarchical_alignment_tool


HAT - Hierachical Alignment Tool
File
Edit
View
Help
Version 1
Обратно на предишния оратор подкрепям изменението на бюджета .
Im Gegensatz zu meinem Vorredner unterstütze ich den Berichtigungshaushalt .
Σε αντιδιαστολή με τον προηγούμενο ομιλητή , εγώ ενέκρινα την τροποποίηση του προϋπολογισμού .
In contradistinction to the previous speaker , I endorsed the amending budget .
A diferencia del ponente anterior , respaldo el presupuesto rectificativo .
Erinevalt eelmisest sõnavõtjast toetan paranduseelarvet .
Päinvastoin kuin edellinen puhuja , annoin tukeni lisätalousarviolle .
À l' inverse de l' orateur précédent , j' ai soutenu la proposition de budget rectificatif .
A differenza dell' oratore che mi ha preceduto , ho appoggiato il bilancio rettificativo .
In tegenstelling tot de vorige spreker heb ik mijn steun verleend aan de gewijzigde begroting .
Ja , w odróżnieniu od mojego przedmówcy , głosowałem za tą poprawką budżetu .
Ao contrário do orador anterior , apoiei o orçamento rectificativo .
Spre deosebire de antevorbitor , am susținut bugetul rectificativ .
Ja som , na rozdiel od predrečníka , hlasoval za opravný rozpočet .
V nasprotju s predhodnim govornikom sem potrdil spremembo proračuna .
I motsats till föregående talare gav jag mitt stöd till ändringsbudgeten .
föregående/föregå/VERB (1122021403)
Обратно
Im Gegensatz
Σε αντιδιαστολή
In contradistinction
A diferencia
Erinevalt
Päinvastoin
À l' inverse
A differenza
In tegenstelling
w odróżnieniu
Ao contrário
Spre deosebire
na rozdiel
V nasprotju
I motsats
на предишния оратор
zu meinem Vorredner
με τον προηγούμενο ομιλητή
to the previous speaker
del ponente anterior
eelmisest sõnavõtjast
kuin edellinen puhuja
de l' orateur précédent
dell' oratore che mi ha preceduto
tot de vorige spreker
od mojego przedmówcy
do orador anterior
de antevorbitor
od predrečníka
s predhodnim govornikom
till föregående
amending
rectificativo
rectificatif
rettificativo
gewijzigde
poprawką
rectificativo
rectificativ
opravný
spremembo
budget
presupuesto
budget
bilancio
begroting
budżetu
orçamento
bugetul
rozpočet
proračuna
изменението на бюджета
Berichtigungshaushalt
τροποποίηση του προϋπολογισμού
paranduseelarvet
lisätalousarviolle
ändringsbudgeten

Appendix B

Alignment Quality

B.1 Relation of Alignment Error Rate (AER) and F_1-Score

Both the alignment error rate (AER)[1] and the F-Score measure take values in the range from 0 (no error; no correct match) to 1 (only errors; no incorrect match). If no distinction is made between sure and possible alignments (i.e., $\mathcal{G}_{sure} = \mathcal{G}_{possible} = \mathcal{G}$), the inverse AER equals the balanced F_1-Score, which is also known as Dice similarity coefficient. To illustrate this correspondence, we replace the sets $\mathcal{T}$ and $\mathcal{G}$ and the measures precision (P) and recall (R) with their respective definition below. Och and Ney (2003) also note when defining the AER that "[it] is derived from the well-known F-measure".

$$
\begin{aligned}
1 - AER &= F_1 \\
\frac{|\mathcal{T} \cap \mathcal{G}_{sure}| + |\mathcal{T} \cap \mathcal{G}_{possible}|}{|\mathcal{T}| + |\mathcal{G}_{sure}|} &= \frac{2 \cdot P \cdot R}{P + R} \\
\frac{2 \cdot |\mathcal{T} \cap \mathcal{G}|}{|\mathcal{T}| + |\mathcal{G}|} &= \frac{2 \cdot \frac{TP}{TP+FP} \cdot \frac{TP}{TP+FN}}{\frac{TP}{TP+FP} + \frac{TP}{TP+FN}} \\
\frac{2 \cdot TP}{(TP + FP) + (TP + FN)} &= \frac{\frac{2 \cdot TP^2}{(TP+FP) \cdot (TP+FN)}}{\frac{TP \cdot (TP+FN) + TP \cdot (TP+FP)}{(TP+FP) \cdot (TP+FN)}} \\
\frac{2 \cdot TP}{2 \cdot TP + FP + FN} &= \frac{2 \cdot TP^2}{TP \cdot (TP + FN) + TP \cdot (TP + FP)} \\
&= \frac{2 \cdot TP}{2 \cdot TP + FP + FN}
\end{aligned}
$$

[1] Originally defined by (Och and Ney 2003) for two different types of alignments: 'sure' and 'possible' ones.

Appendix C

Data Sets from Joint Measures

C.1 Semantic Relatedness of German Particle Verbs

We aim at calculating how different the meaning of a particle verb is from its base verb (see Section 3.2.2). For this purpose, we use the lemma alignment distribution overlap function O_a described in Section 5.1, which takes into account the probabilities of all lemmas λ_x that occur in the lemma distribution matrix explained in Section 3.2.1 for both base verb and particle verb consisting of base verb and prefix particle. O_a, which is the sum of the respective lower probabilities, yields a value between 0 and 1, which denotes the percentage of foreign language (i.e., not German) lemmas that serve as a translation by means of word alignments (see Section 4.4) of both verbs.

In total, we identify 1592 different particle verbs in the German part of our corpus. For 312 of them, we find no overlap, that is no single common lemma that has been assigned to tokens aligned with both verbs. The following list comprises only those 745 particle verbs that show an absolute overlap frequency of at least 10 in at least three languages.

We count these frequencies as the sum of constellations where one of the verbs and the lemma of a foreign word occur, which can be in the extreme cases three lemma pairs that occur together 10 times or 10 different lemmas that occur once, each in a different language.

Base verb	Particle verb	O_a	Frequency
lösen	auslösen	0.1 %	12
halten	innehalten	0.1 %	10
schaffen	abschaffen	0.2 %	18
arbeiten	herausarbeiten	0.2 %	10
lassen	freilassen	0.3 %	28
führen	irreführen	0.7 %	31
stellen	klarstellen	0.7 %	89

Base verb	Particle verb	O_a	Frequency
handeln	aushandeln	0.7 %	44
ziehen	vorziehen	0.7 %	18
schließen	einschließen	0.8 %	80
leiten	weiterleiten	0.9 %	12
bringen	durcheinanderbringen	1.0 %	10
hören	aufhören	1.1 %	135
geben	übergeben	1.1 %	40
sehen	durchsehen	1.1 %	13
bringen	anbringen	1.1 %	57
sehen	absehen	1.1 %	133
sehen	wegsehen	1.1 %	12
machen	weitermachen	1.1 %	104
schließen	zusammenschließen	1.1 %	53
sprechen	absprechen	1.1 %	11
stellen	sicherstellen	1.2 %	203
legen	auslegen	1.3 %	24
greifen	vorgreifen	1.3 %	10
setzen	aussetzen	1.3 %	91
setzen	absetzen	1.4 %	12
nehmen	vorwegnehmen	1.4 %	11
weisen	anweisen	1.4 %	15
wenden	einwenden	1.4 %	12
nehmen	hinnehmen	1.5 %	152
führen	ausführen	1.5 %	165
führen	weiterführen	1.5 %	72
führen	aufführen	1.6 %	12
werfen	vorwerfen	1.6 %	12
wirken	entgegenwirken	1.7 %	14
nehmen	ausnehmen	1.8 %	21
schreiben	festschreiben	1.8 %	21
bringen	überbringen	1.8 %	21
messen	zumessen	1.8 %	40
greifen	angreifen	1.9 %	16
setzen	fortsetzen	1.9 %	180

Base verb	Particle verb	O_a	Frequency
lassen	nachlassen	1.9 %	36
messen	beimessen	2.0 %	60
führen	zusammenführen	2.0 %	15
führen	fortführen	2.0 %	90
stellen	voranstellen	2.1 %	14
bringen	entgegenbringen	2.2 %	15
gehen	vorübergehen	2.2 %	18
wenden	anwenden	2.2 %	118
nehmen	hinzunehmen	2.2 %	41
zögern	hinauszögern	2.2 %	24
beziehen	einbeziehen	2.3 %	275
sehen	vorsehen	2.3 %	1086
stellen	gleichstellen	2.4 %	11
geben	mitgeben	2.5 %	14
setzen	auseinandersetzen	2.7 %	37
teilen	mitteilen	2.7 %	231
stellen	entgegenstellen	2.8 %	10
treten	zurücktreten	2.8 %	14
sehen	voraussehen	2.8 %	247
setzen	voraussetzen	2.8 %	186
halten	aufhalten	2.9 %	158
lenken	ablenken	2.9 %	36
tragen	vortragen	2.9 %	45
setzen	zusammensetzen	3.0 %	90
teilen	zuteilen	3.1 %	15
teilen	einteilen	3.1 %	35
machen	klarmachen	3.3 %	243
stellen	ausstellen	3.3 %	38
lösen	loslösen	3.5 %	21
sagen	zusagen	3.5 %	114
kommen	umkommen	3.5 %	20
nehmen	zurücknehmen	3.5 %	137
stellen	herstellen	3.7 %	322
treten	zusammentreten	3.7 %	39

Base verb	Particle verb	O_a	Frequency
stimmen	übereinstimmen	3.7 %	2090
setzen	durchsetzen	3.8 %	346
arbeiten	ausarbeiten	3.8 %	357
leiten	einleiten	3.8 %	75
machen	durchmachen	3.8 %	39
geben	zugeben	3.8 %	415
stellen	einstellen	3.9 %	147
geben	ausgeben	3.9 %	243
nehmen	vornehmen	3.9 %	257
treffen	zutreffen	3.9 %	377
geben	freigeben	4.0 %	21
geben	zurückgeben	4.2 %	40
schließen	ausschließen	4.2 %	388
stellen	feststellen	4.2 %	1472
finden	abfinden	4.3 %	11
kommen	nachkommen	4.3 %	176
werten	auswerten	4.4 %	11
machen	vormachen	4.5 %	82
führen	einführen	4.6 %	941
gehen	nachgehen	4.7 %	66
halten	mithalten	4.9 %	41
kommen	vorankommen	4.9 %	559
gehen	weggehen	4.9 %	48
stellen	vorstellen	4.9 %	848
machen	mitmachen	4.9 %	29
binden	einbinden	5.1 %	49
nehmen	annehmen	5.1 %	1863
reichen	zurückreichen	5.2 %	15
weisen	abweisen	5.2 %	49
setzen	umsetzen	5.2 %	1216
geben	preisgeben	5.2 %	11
greifen	herausgreifen	5.2 %	26
fordern	zurückfordern	5.4 %	41
denken	überdenken	5.4 %	529

Base verb	Particle verb	O_a	Frequency
halten	anhalten	5.4 %	111
schreiben	vorschreiben	5.4 %	79
brechen	zusammenbrechen	5.5 %	49
weisen	zuweisen	5.5 %	32
stellen	herausstellen	5.5 %	232
geben	weitergeben	5.5 %	114
kommen	vorwärtskommen	5.6 %	15
wirken	einwirken	5.6 %	15
treffen	eintreffen	5.6 %	46
fangen	anfangen	5.6 %	50
stellen	bereitstellen	5.6 %	564
nehmen	zunehmen	5.7 %	443
wirken	mitwirken	5.8 %	91
sagen	voraussagen	5.8 %	46
fordern	überfordern	5.8 %	22
fallen	zusammenfallen	5.8 %	30
bringen	voranbringen	5.8 %	218
gehen	herangehen	5.9 %	89
knüpfen	anknüpfen	5.9 %	16
legen	einlegen	6.0 %	21
geben	herausgeben	6.0 %	53
schließen	anschließen	6.1 %	773
verteilen	umverteilen	6.1 %	77
führen	zurückführen	6.2 %	449
fassen	zusammenfassen	6.3 %	65
halten	standhalten	6.4 %	49
legen	offenlegen	6.5 %	20
stellen	zurückstellen	6.5 %	10
gehen	umgehen	6.5 %	645
reichen	weiterreichen	6.6 %	12
setzen	einsetzen	6.6 %	1080
klagen	anklagen	6.6 %	35
rufen	hervorrufen	6.7 %	111
bringen	zusammenbringen	6.7 %	84

Base verb	Particle verb	O_a	Frequency
gehen	vorausgehen	6.8 %	60
halten	zusammenhalten	6.8 %	41
kommen	auskommen	6.8 %	33
fallen	auffallen	6.9 %	29
streiten	abstreiten	6.9 %	16
geben	angeben	7.0 %	170
legen	ablegen	7.0 %	17
legen	anlegen	7.0 %	48
halten	zurückhalten	7.1 %	75
liefern	ausliefern	7.1 %	125
setzen	hinsetzen	7.1 %	31
geben	bekanntgeben	7.4 %	63
schieben	abschieben	7.5 %	11
greifen	eingreifen	7.5 %	85
kommen	zurückkommen	7.6 %	659
sehen	gegenübersehen	7.7 %	309
kommen	überkommen	7.7 %	13
treten	auftreten	7.7 %	239
streichen	zusammenstreichen	7.8 %	13
tragen	mittragen	7.8 %	149
finden	zurückfinden	7.8 %	34
halten	durchhalten	7.9 %	26
nehmen	abnehmen	7.9 %	133
nehmen	teilnehmen	8.0 %	810
treten	beitreten	8.0 %	509
setzen	entgegensetzen	8.1 %	19
weisen	zurückweisen	8.1 %	128
nehmen	herausnehmen	8.1 %	100
lassen	einlassen	8.1 %	20
machen	ausmachen	8.2 %	328
fordern	herausfordern	8.2 %	82
stellen	anstellen	8.4 %	78
halten	einhalten	8.5 %	3412
halten	abhalten	8.7 %	903

Base verb	Particle verb	O_a	Frequency
kommen	wiederkommen	8.7 %	29
laufen	anlaufen	8.8 %	123
halten	heraushalten	8.9 %	12
laufen	weiterlaufen	8.9 %	33
tragen	beitragen	8.9 %	1434
greifen	aufgreifen	8.9 %	109
wandeln	umwandeln	8.9 %	39
fügen	beifügen	8.9 %	14
gehen	übergehen	9.0 %	527
kommen	zusammenkommen	9.0 %	105
stellen	zusammenstellen	9.0 %	25
legen	darlegen	9.0 %	333
zeichnen	auszeichnen	9.0 %	23
schlagen	niederschlagen	9.0 %	23
lassen	auslassen	9.1 %	43
fallen	ausfallen	9.1 %	158
hören	abhören	9.1 %	18
brechen	einbrechen	9.2 %	15
fahren	fortfahren	9.3 %	147
gehen	untergehen	9.3 %	43
ziehen	abziehen	9.3 %	76
gehen	abgehen	9.5 %	17
weisen	ausweisen	9.6 %	97
stellen	abstellen	9.6 %	44
gehen	zurückgehen	9.6 %	401
kommen	dazukommen	9.6 %	15
räumen	einräumen	9.7 %	30
führen	anführen	9.8 %	227
kommen	weiterkommen	9.9 %	174
fallen	wegfallen	9.9 %	41
bringen	einbringen	10.0 %	809
kommen	zurechtkommen	10.0 %	24
sagen	absagen	10.0 %	27
stimmen	zustimmen	10.0 %	6221

Base verb	Particle verb	O_a	Frequency
drücken	ausdrücken	10.1 %	63
führen	herbeiführen	10.1 %	282
gehen	sichergehen	10.1 %	16
nehmen	aufnehmen	10.2 %	1669
bereiten	vorbereiten	10.2 %	248
bringen	vorbringen	10.5 %	322
leiten	zuleiten	10.6 %	10
kommen	entgegenkommen	10.7 %	69
gehen	verlorengehen	10.8 %	74
geben	aufgeben	10.9 %	408
drängen	aufdrängen	10.9 %	39
fallen	anfallen	10.9 %	16
richten	einrichten	10.9 %	170
eignen	aneignen	10.9 %	26
laufen	hinauslaufen	10.9 %	141
legen	nahelegen	10.9 %	67
wechseln	abwechseln	11.0 %	12
bauen	ausbauen	11.0 %	218
reichen	hinausreichen	11.0 %	29
weisen	nachweisen	11.1 %	140
geben	vorgeben	11.1 %	186
behalten	vorbehalten	11.1 %	95
bringen	aufbringen	11.2 %	67
fallen	einfallen	11.2 %	23
halten	festhalten	11.2 %	1624
legen	festlegen	11.2 %	578
stehen	zusammenstehen	11.2 %	16
legen	niederlegen	11.3 %	66
handeln	abhandeln	11.4 %	34
kommen	aufkommen	11.4 %	178
stellen	aufstellen	11.4 %	305
sammeln	ansammeln	11.4 %	38
führen	heranführen	11.6 %	33
finden	stattfinden	11.7 %	5101

Base verb	Particle verb	O_a	Frequency
gehen	vorgehen	11.7 %	775
finden	zusammenfinden	11.7 %	21
bringen	hervorbringen	11.9 %	242
werfen	aufwerfen	11.9 %	182
geben	nachgeben	12.0 %	192
bereiten	zubereiten	12.0 %	10
schlagen	zuschlagen	12.1 %	23
kommen	durchkommen	12.1 %	66
zwingen	aufzwingen	12.2 %	511
nehmen	wegnehmen	12.3 %	190
sprechen	aussprechen	12.3 %	1482
setzen	ansetzen	12.4 %	91
kommen	übereinkommen	12.4 %	95
rüsten	aufrüsten	12.5 %	22
laufen	herumlaufen	12.6 %	14
bilden	ausbilden	12.7 %	405
passen	anpassen	12.8 %	238
fassen	auffassen	12.8 %	13
rufen	zurückrufen	12.8 %	13
formulieren	umformulieren	12.9 %	77
stimmen	einstimmen	12.9 %	21
halten	aushalten	13.0 %	29
laufen	auslaufen	13.0 %	292
nehmen	wahrnehmen	13.1 %	494
räumen	aufräumen	13.2 %	14
legen	auflegen	13.4 %	19
wenden	abwenden	13.4 %	63
sperren	einsperren	13.5 %	12
merken	anmerken	13.8 %	146
kommen	gleichkommen	13.8 %	326
stellen	gegenüberstellen	13.9 %	25
gehen	hervorgehen	13.9 %	474
laufen	zuwiderlaufen	14.0 %	150
gehen	eingehen	14.0 %	2465

Base verb	Particle verb	O_a	Frequency
sprechen	ansprechen	14.0 %	5335
gehen	ausgehen	14.0 %	1562
kommen	zukommen	14.1 %	1112
zeichnen	abzeichnen	14.2 %	29
heben	abheben	14.2 %	10
geben	abgeben	14.2 %	1018
drehen	umdrehen	14.2 %	15
weisen	aufweisen	14.3 %	178
stürzen	einstürzen	14.4 %	18
finden	herausfinden	14.5 %	411
kehren	zurückkehren	14.5 %	48
führen	durchführen	14.6 %	2502
gehen	aufgehen	14.7 %	43
gehen	durchgehen	14.7 %	74
reißen	niederreißen	14.7 %	10
bringen	näherbringen	14.7 %	93
sehen	aussehen	14.7 %	1732
prüfen	nachprüfen	14.8 %	204
dringen	eindringen	14.8 %	13
lösen	auflösen	14.8 %	62
fließen	zurückfließen	14.9 %	12
denken	ausdenken	15.0 %	55
gestalten	ausgestalten	15.1 %	66
erhalten	aufrechterhalten	15.1 %	3221
sprechen	zusprechen	15.1 %	17
bringen	herausbringen	15.1 %	16
gehen	angehen	15.2 %	4423
zeigen	anzeigen	15.2 %	202
lassen	hinterlassen	15.4 %	693
stellen	darstellen	15.7 %	4413
dehnen	ausdehnen	15.7 %	11
denken	andenken	15.7 %	36
gehen	weitergehen	15.7 %	1002
stecken	feststecken	15.8 %	17

Base verb	Particle verb	O_a	Frequency
kommen	herauskommen	15.9 %	232
drängen	zurückdrängen	15.9 %	13
rufen	ausrufen	15.9 %	34
stellen	hinstellen	16.0 %	25
rechnen	ausrechnen	16.2 %	50
streichen	herausstreichen	16.2 %	11
stehen	aufstehen	16.2 %	72
weichen	abweichen	16.2 %	18
geben	stattgeben	16.2 %	20
sprechen	herumsprechen	16.3 %	10
legen	vorlegen	16.4 %	1099
rechnen	nachrechnen	16.6 %	17
führen	vorbeiführen	16.6 %	16
sparen	aussparen	16.7 %	16
kommen	hinkommen	16.7 %	114
gehen	zugehen	16.9 %	71
kommen	rauskommen	17.0 %	13
gehen	einhergehen	17.1 %	356
nehmen	einnehmen	17.1 %	740
bringen	weiterbringen	17.1 %	75
stehen	einstehen	17.2 %	44
gehen	vorangehen	17.3 %	282
lassen	zurücklassen	17.4 %	96
heben	herausheben	17.5 %	18
sehen	einsehen	17.6 %	1626
teilen	aufteilen	17.7 %	756
nehmen	entgegennehmen	17.7 %	103
enthalten	vorenthalten	17.8 %	57
scheuen	zurückscheuen	17.9 %	17
kommen	vorkommen	17.9 %	1030
ordnen	einordnen	18.2 %	10
laufen	ablaufen	18.2 %	395
gewinnen	zurückgewinnen	18.2 %	75
rufen	anrufen	18.2 %	90

Base verb	Particle verb	O_a	Frequency
erhalten	zurückerhalten	18.4 %	30
bewahren	aufbewahren	18.6 %	128
treten	eintreten	19.0 %	588
ziehen	zurückziehen	19.0 %	698
fließen	einfließen	19.2 %	10
bekommen	zurückbekommen	19.8 %	21
schätzen	einschätzen	19.9 %	335
brechen	auseinanderbrechen	19.9 %	18
lassen	vorlassen	19.9 %	14
segnen	absegnen	20.0 %	12
denken	nachdenken	20.0 %	3862
holen	einholen	20.0 %	13
sehen	zusehen	20.2 %	341
zahlen	zurückzahlen	20.3 %	80
stehen	vorstehen	20.3 %	18
rufen	zurufen	20.3 %	37
treten	antreten	20.4 %	34
sterben	aussterben	20.5 %	37
stehen	zustehen	20.6 %	270
kehren	umkehren	20.8 %	31
steuern	zusteuern	21.0 %	24
holen	abholen	21.1 %	12
gehören	zugehören	21.2 %	11
fahren	hinfahren	21.3 %	14
geben	hingeben	21.3 %	32
laden	aufladen	21.3 %	11
tun	abtun	21.3 %	14
ankommen	herankommen	21.4 %	15
deuten	andeuten	21.5 %	62
rütteln	aufrütteln	21.5 %	13
schreiten	fortschreiten	21.6 %	18
melden	anmelden	21.9 %	121
reisen	einreisen	22.0 %	120
hängen	aufhängen	22.1 %	38

Base verb	Particle verb	O_a	Frequency
fügen	hinzufügen	22.1 %	26
bestehen	fortbestehen	22.3 %	262
spielen	abspielen	22.5 %	84
weisen	hinweisen	22.8 %	898
sehen	weitersehen	22.8 %	15
sehen	ansehen	22.9 %	4810
halten	auseinanderhalten	23.0 %	21
gehen	hinausgehen	23.1 %	1745
kommen	hereinkommen	23.2 %	44
verweisen	zurückverweisen	23.2 %	152
bleiben	ausbleiben	23.3 %	32
laufen	hinterherlaufen	23.4 %	16
rücken	näherrücken	23.4 %	30
kommen	herankommen	23.4 %	17
liefern	abliefern	23.5 %	22
stürzen	abstürzen	23.5 %	26
gehen	losgehen	23.5 %	24
stehen	draufstehen	23.6 %	16
erstatten	zurückerstatten	23.8 %	53
bieten	überbieten	23.8 %	17
brechen	abbrechen	23.9 %	86
laden	einladen	24.0 %	40
nehmen	übernehmen	24.0 %	3942
gestalten	mitgestalten	24.1 %	67
geben	hergeben	24.2 %	25
gehen	vorbeigehen	24.2 %	77
lassen	hereinlassen	24.3 %	19
halten	vorhalten	24.3 %	11
gehen	auseinandergehen	24.3 %	51
fügen	anfügen	24.4 %	17
holen	herausholen	24.4 %	24
schieben	zuschieben	24.5 %	24
bringen	zurückbringen	24.6 %	11
heben	anheben	24.9 %	51

Base verb	Particle verb	O_a	Frequency
gehen	mitgehen	24.9 %	15
regen	anregen	24.9 %	11
lassen	offenlassen	25.1 %	73
tun	antun	25.1 %	24
gehören	hingehören	25.1 %	64
rechnen	anrechnen	25.1 %	10
gehen	hingehen	25.1 %	92
kommen	hineinkommen	25.6 %	12
treiben	vorantreiben	25.7 %	155
heben	aufheben	25.7 %	54
landen	anlanden	25.9 %	61
sehen	entgegensehen	26.0 %	706
kommen	nahekommen	26.1 %	23
bleiben	zusammenbleiben	26.1 %	17
herrschen	vorherrschen	26.1 %	320
behandeln	gleichbehandeln	26.2 %	86
halten	bereithalten	26.2 %	39
wirken	hinwirken	26.4 %	80
billigen	zubilligen	26.5 %	10
blicken	zurückblicken	26.5 %	60
lassen	übriglassen	26.6 %	303
schieben	aufschieben	26.9 %	50
finden	zurechtfinden	27.3 %	10
finden	bereitfinden	27.3 %	25
bestimmen	mitbestimmen	27.5 %	18
schieben	hinausschieben	27.6 %	40
setzen	festsetzen	27.8 %	644
stehen	freistehen	27.9 %	35
zeichnen	vorzeichnen	28.1 %	23
richten	ausrichten	28.1 %	922
entscheiden	mitentscheiden	28.1 %	37
gehören	zusammengehören	28.1 %	126
zählen	aufzählen	28.1 %	101
sehen	umsehen	28.2 %	47

Base verb	Particle verb	O_a	Frequency
treffen	zusammentreffen	28.4 %	932
arbeiten	zusammenarbeiten	28.5 %	7519
wünschen	herbeiwünschen	28.6 %	13
kommen	vorbeikommen	28.6 %	20
schließen	abschließen	28.7 %	6451
werfen	zurückwerfen	28.8 %	37
lassen	zulassen	28.8 %	5310
fressen	auffressen	28.9 %	13
zählen	auszählen	28.9 %	13
sitzen	festsitzen	28.9 %	20
fordern	abfordern	28.9 %	20
erinnern	zurückerinnern	29.0 %	12
fließen	abfließen	29.0 %	28
hören	hinhören	29.1 %	33
kündigen	ankündigen	29.3 %	31
kommen	ankommen	29.6 %	2386
kommen	zugutekommen	29.7 %	249
stehen	beistehen	29.7 %	61
gehen	fortgehen	29.8 %	15
bauen	einbauen	29.9 %	29
heben	hervorheben	30.3 %	90
liegen	vorliegen	30.3 %	3540
ordnen	anordnen	30.3 %	28
schlagen	zusammenschlagen	30.5 %	124
löschen	auslöschen	30.5 %	56
fallen	zurückfallen	30.5 %	40
legen	drauflegen	30.6 %	13
schreiten	voranschreiten	30.6 %	50
stehen	drinstehen	30.6 %	18
dauern	andauern	30.8 %	338
bestehen	weiterbestehen	30.9 %	90
vertrauen	anvertrauen	30.9 %	84
kommen	näherkommen	30.9 %	40
gestehen	zugestehen	30.9 %	139

Base verb	Particle verb	O_a	Frequency
schlafen	einschlafen	31.0 %	42
raten	abraten	31.1 %	51
greifen	zurückgreifen	31.2 %	253
spielen	mitspielen	31.3 %	41
nehmen	mitnehmen	31.4 %	81
stehen	nachstehen	31.4 %	22
gehen	entgegengehen	31.5 %	27
wenden	zuwenden	31.6 %	236
wirken	auswirken	31.8 %	330
fahren	weiterfahren	31.8 %	10
passen	zusammenpassen	31.9 %	47
kommen	herkommen	31.9 %	275
kommen	zustandekommen	32.1 %	200
schweigen	ausschweigen	33.1 %	64
mahnen	anmahnen	33.2 %	58
sehen	hinsehen	33.6 %	16
gehören	angehören	33.8 %	3592
handeln	zuwiderhandeln	33.9 %	15
liegen	zurückliegen	33.9 %	153
leben	aufleben	34.1 %	10
sitzen	einsitzen	34.3 %	13
gehen	herausgehen	34.4 %	11
kommen	hinauskommen	34.7 %	11
schicken	zurückschicken	34.8 %	164
bleiben	zurückbleiben	34.8 %	262
klären	aufklären	34.9 %	461
hören	zuhören	35.0 %	4553
hören	heraushören	35.0 %	13
kommen	herumkommen	35.2 %	35
nutzen	ausnutzen	35.2 %	1738
häufen	anhäufen	35.2 %	54
mischen	einmischen	35.5 %	31
bewegen	zubewegen	35.5 %	109
fordern	nachfordern	35.6 %	10

Base verb	Particle verb	O_a	Frequency
passen	hineinpassen	35.7 %	11
arbeiten	mitarbeiten	35.8 %	597
weichen	zurückweichen	36.0 %	15
arbeiten	abarbeiten	36.0 %	25
folgen	nachfolgen	36.2 %	19
bringen	mitbringen	36.4 %	147
schlachten	abschlachten	36.5 %	27
sprechen	dagegensprechen	36.5 %	12
stehen	entgegenstehen	36.8 %	167
zahlen	auszahlen	36.9 %	798
kündigen	aufkündigen	36.9 %	19
kommen	daherkommen	37.0 %	45
gehören	hineingehören	37.0 %	56
kommen	hinzukommen	37.2 %	462
bilden	herausbilden	37.2 %	28
wandern	abwandern	37.3 %	13
packen	anpacken	37.5 %	33
halten	hinhalten	37.6 %	16
liegen	beiliegen	37.6 %	11
fließen	zufließen	37.7 %	21
spielen	ausspielen	37.8 %	108
bleiben	dableiben	38.1 %	11
rufen	aufrufen	38.3 %	623
weiten	ausweiten	38.3 %	25
rüsten	ausrüsten	38.5 %	214
geben	widergeben	38.5 %	22
sichern	zusichern	38.6 %	855
fordern	anfordern	38.8 %	429
fügen	einfügen	38.9 %	56
fangen	auffangen	38.9 %	24
führen	hinführen	39.3 %	25
hängen	zusammenhängen	39.6 %	328
sprechen	mitsprechen	39.9 %	10
sagen	ansagen	40.3 %	51

Base verb	Particle verb	O_a	Frequency
schauen	zuschauen	40.3 %	124
räumen	ausräumen	40.4 %	179
wachsen	nachwachsen	40.6 %	14
brennen	niederbrennen	40.7 %	58
rechnen	mitrechnen	40.7 %	15
stehen	ausstehen	40.8 %	167
rühren	herrühren	40.8 %	51
stehen	offenstehen	41.2 %	179
hören	anhören	41.2 %	2134
verkaufen	weiterverkaufen	41.2 %	38
tun	auftun	41.4 %	15
wachsen	aufwachsen	41.5 %	432
denken	zurückdenken	41.6 %	72
klären	abklären	41.6 %	29
halten	zugutehalten	41.6 %	24
schießen	hinausschießen	42.4 %	21
erkannt	anerkannt	42.5 %	34
sammeln	einsammeln	42.6 %	48
stammen	herstammen	42.7 %	11
wachsen	weiterwachsen	42.7 %	27
kommen	hierherkommen	42.8 %	74
bessern	nachbessern	42.9 %	105
erkennen	zuerkennen	43.0 %	123
verfolgen	weiterverfolgen	43.1 %	529
stehen	bevorstehen	43.2 %	385
erkennen	anerkennen	43.3 %	7006
schauen	zurückschauen	43.4 %	29
trocknen	austrocknen	43.5 %	12
liegen	ausliegen	43.5 %	16
blühen	aufblühen	43.6 %	38
spielen	zuspielen	43.8 %	12
überweisen	zurücküberweisen	43.8 %	31
schauen	ausschauen	43.9 %	25
schauen	wegschauen	44.1 %	40

Base verb	Particle verb	O_a	Frequency
schauen	umschauen	44.5 %	68
schießen	abschießen	44.6 %	12
gehören	dazugehören	45.1 %	159
zweifeln	anzweifeln	45.1 %	131
brechen	aufbrechen	45.3 %	40
erleben	miterleben	45.6 %	192
werfen	hinauswerfen	45.8 %	19
senken	absenken	45.8 %	111
zielen	hinzielen	45.8 %	19
treiben	antreiben	45.9 %	22
brennen	abbrennen	46.0 %	39
prangern	anprangern	46.4 %	24
verlangen	zurückverlangen	46.4 %	18
atmen	aufatmen	46.5 %	22
denken	weiterdenken	46.5 %	32
geben	eingeben	46.7 %	13
schwächen	abschwächen	46.7 %	490
zählen	mitzählen	46.7 %	44
hängen	abhängen	46.8 %	400
wünschen	zurückwünschen	47.0 %	11
wählen	auswählen	47.3 %	1548
stehen	nahestehen	47.3 %	16
stehen	gegenüberstehen	47.4 %	2148
liegen	festliegen	47.5 %	14
finanzieren	mitfinanzieren	47.8 %	21
deuten	hindeuten	47.8 %	121
betreffen	anbetreffen	47.9 %	297
denken	vordenken	48.2 %	32
wachsen	heranwachsen	48.5 %	21
stehen	anstehen	48.5 %	319
wachsen	zusammenwachsen	48.5 %	105
springen	überspringen	48.6 %	22
breiten	ausbreiten	48.7 %	14
brechen	ausbrechen	48.8 %	180

Base verb	Particle verb	O_a	Frequency
handeln	einhandeln	49.0 %	11
ändern	abändern	49.0 %	2092
sinken	absinken	49.3 %	52
stehen	bereitstehen	49.4 %	235
planen	einplanen	49.6 %	16
trauen	zutrauen	49.7 %	74
scheinen	aufscheinen	49.8 %	10
verfolgen	mitverfolgen	49.8 %	44
mildern	abmildern	49.9 %	167
stehen	feststehen	50.0 %	936
tauschen	austauschen	50.4 %	53
behalten	beibehalten	50.5 %	2031
hinken	hinterherhinken	50.6 %	26
schlagen	vorschlagen	50.8 %	659
erklären	bereiterklären	50.9 %	85
liegen	bereitliegen	51.1 %	42
üben	ausüben	51.2 %	98
bleiben	gleichbleiben	51.4 %	24
kaufen	einkaufen	51.5 %	464
fordern	auffordern	51.6 %	28 604
wachsen	anwachsen	51.8 %	219
fallen	herausfallen	51.9 %	19
sichern	absichern	52.0 %	354
bleiben	übrigbleiben	52.1 %	317
gestehen	eingestehen	52.2 %	596
leben	zusammenleben	53.1 %	417
formulieren	ausformulieren	53.1 %	32
entwickeln	fortentwickeln	53.1 %	47
gewöhnen	angewöhnen	53.2 %	38
schieben	herschieben	53.4 %	27
sprechen	vorsprechen	53.5 %	18
hungern	aushungern	53.7 %	47
zeigen	aufzeigen	54.0 %	3306
schauen	hinschauen	54.4 %	30

Base verb	Particle verb	O_a	Frequency
fallen	zufallen	54.4 %	11
leben	weiterleben	54.5 %	23
liegen	auseinanderliegen	54.6 %	37
locken	anlocken	54.9 %	69
ruhen	ausruhen	55.2 %	50
sitzen	gegenübersitzen	55.8 %	12
decken	abdecken	56.0 %	1419
gehen	zugrundegehen	56.8 %	13
füllen	ausfüllen	57.7 %	190
nähern	annähern	57.9 %	486
fordern	einfordern	57.9 %	1551
finden	vorfinden	58.0 %	165
bremsen	ausbremsen	58.4 %	16
denken	hinausdenken	58.4 %	31
stehen	dastehen	58.7 %	236
nicken	abnicken	58.9 %	12
sterben	absterben	58.9 %	29
bieten	anbieten	59.0 %	6548
schränken	einschränken	59.0 %	26
liegen	naheliegen	59.1 %	51
lachen	auslachen	59.2 %	35
verlangen	abverlangen	59.4 %	240
schreiben	abschreiben	59.5 %	26
pflichten	beipflichten	59.5 %	52
bezahlen	zurückbezahlen	59.9 %	16
reifen	heranreifen	60.0 %	30
fragen	nachfragen	60.6 %	183
sitzen	dasitzen	60.6 %	22
schauen	anschauen	60.7 %	1344
helfen	mithelfen	61.2 %	669
probieren	ausprobieren	61.3 %	53
drohen	androhen	61.5 %	180
arbeiten	weiterarbeiten	61.7 %	442
grenzen	angrenzen	62.5 %	61

Base verb	Particle verb	O_a	Frequency
sitzen	herumsitzen	63.0 %	19
gewinnen	hinzugewinnen	63.5 %	13
bauen	aufbauen	63.5 %	3744
lehnen	ablehnen	63.7 %	733
bleiben	offenbleiben	64.0 %	125
wissen	weiterwissen	64.2 %	46
dringlich	vordringlich	64.5 %	453
sitzen	zusammensitzen	64.9 %	28
warten	abwarten	64.9 %	3974
schreiben	anschreiben	65.4 %	64
schreiben	hineinschreiben	65.5 %	54
lernen	auslernen	65.5 %	26
liegen	zugrundeliegen	65.9 %	345
kämpfen	ankämpfen	66.6 %	102
erlegen	auferlegen	67.1 %	10
sagen	aufsagen	67.4 %	33
frieren	einfrieren	67.5 %	10
zahlen	einzahlen	67.8 %	131
streben	anstreben	67.8 %	1211
arbeiten	hinarbeiten	67.8 %	1207
lesen	weiterlesen	67.8 %	10
sprechen	weitersprechen	67.9 %	22
sagen	nachsagen	69.0 %	11
fliegen	anfliegen	69.0 %	43
kämpfen	weiterkämpfen	69.5 %	31
bessern	aufbessern	69.9 %	10
stimmen	mitstimmen	70.2 %	32
kaufen	aufkaufen	70.3 %	205
wechseln	auswechseln	70.4 %	13
stimmen	abstimmen	70.5 %	22 203
schicken	hinschicken	70.7 %	16
sagen	vorsagen	70.8 %	32
reichen	ausreichen	70.8 %	3140
sagen	aussagen	71.0 %	573

Base verb	Particle verb	O_a	Frequency
reihen	einreihen	71.2 %	15
helfen	weiterhelfen	71.4 %	214
lagern	einlagern	71.5 %	59
stimmen	dagegenstimmen	72.5 %	321
senden	zusenden	73.2 %	119
lesen	nachlesen	73.5 %	328
beuten	ausbeuten	73.5 %	42
schicken	losschicken	74.0 %	10
lesen	durchlesen	74.0 %	207
fragen	anfragen	74.1 %	128
schicken	zuschicken	74.3 %	22
zielen	abzielen	74.8 %	754
plündern	ausplündern	75.5 %	146
sparen	einsparen	75.7 %	1327
kennen	auskennen	76.1 %	212
säen	aussäen	77.5 %	27
entwickeln	weiterentwickeln	77.6 %	4127
drucken	abdrucken	78.0 %	16
spionieren	ausspionieren	79.2 %	13
lesen	vorlesen	80.1 %	1126
pflanzen	anpflanzen	81.2 %	29
senden	aussenden	83.1 %	953
spiegeln	widerspiegeln	85.7 %	425
steigen	ansteigen	89.1 %	2833

C.2 Generated Recommendations for Learners of English of Different L1 Backgrounds

Our approach to predict transfer errors in English with regard to preposition use in combination with verbs (VPC) and adjectives (APC), which is detailed in Section 5.4, generates the following lists. The backtranslation ratio (BTR) is an indicator for how probable it is that the wrong (backtranslated) preposition $\lambda_{p''}$ is used instead of the correct one (λ_p). The ranking of the particular combinations has been done on both BTR and the frequency of the respective VPC or APC in our corpus (FEP6, see Chapter 3). To not generate endless lists, we limit the number of recommendations to approximately 100 for VPC and 25 for APC.

C.2.1 Verb Preposition Combinations

	German				French			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
1	think	of	on	1.09	deal	with	of	2.07
2	impose	on	for	1.36	provide	for	of	1.24
3	hope	for	on	1.07	call	for	of	1.82
4	remind	of	on	1.22	decide	on	of	1.05
5	prevent	from	of	1.83	comply	with	of	1.60
6	consist	of	from	1.38	hope	for	of	1.00
7	postpone	until	by	1.06	ask	for	of	2.08
8	exclude	from	of	1.64	face	with	in	1.65
9	aim	at	on	2.74	push	for	of	1.07
10	talk	about	on	3.34	confront	with	in	1.19
11	look	at	in	3.40	cope	with	in	1.47
12	gain	from	of	1.42	reserve	for	in	1.19
13	deliver	on	in	1.37	inflict	on	in	1.11
14	receive	from	of	2.00	spend	on	for	1.75
15	emanate	from	of	1.19	apologise	for	of	1.26
16	compose	of	from	1.30	qualify	for	of	1.15
17	wait	for	on	2.25	strive	for	of	1.32
18	embark	on	in	1.69	associate	with	in	1.92
19	compliment	on	for	1.49	wait	for	of	1.99
20	benefit	from	of	2.72	aim	at	in	2.81
21	shed	on	in	1.62	last	for	of	1.23

	German				French			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
22	suffer	from	under	2.44	expire	on	in	1.25
23	dispense	with	on	1.57	allow	for	of	2.28
24	stop	from	of	1.88	arrange	for	of	1.44
25	warn	against	before	1.82	cater	for	of	1.45
26	protect	from	before	2.42	confer	on	in	1.79
26	protect	from	before	2.42	confer	on	in	1.79
27	test	on	in	1.65	look	at	in	5.08
28	abstain	from	in	2.42	account	for	of	2.50
29	hear	from	of	2.65	arrive	at	in	2.77
30	refrain	from	of	2.44	embark	on	in	2.37
31	inform	of	on	2.92	blame	for	of	2.28
32	profit	from	of	2.14	direct	at	in	2.79
33	free	from	of	2.21	destine	for	in	2.27
34	direct	at	on	2.74	estimate	at	in	2.42
35	spend	on	for	3.76	resume	at	in	2.31
36	target	at	on	2.66	burden	with	of	2.28
37	worry	about	on	3.01	concern	with	of	4.26
38	estimate	at	on	2.49	align	with	on	2.59
39	recover	from	of	2.45	fill	with	of	2.69
40	delight	with	on	2.65	congratulate	on	for	6.68
41	depend	on	of	5.01	depend	on	of	5.77
42	arrive	at	in	4.07	search	for	of	2.98
43	exempt	from	of	3.13	level	at	in	3.04
44	differ	from	of	3.62	please	with	of	4.43
45	level	at	on	2.99	care	for	of	3.59
46	depart	from	of	3.24	dispense	with	of	3.34
47	expect	from	of	3.91	forgive	for	of	4.25
48	complain	about	on	3.55	target	at	on	4.97
49	distance	from	of	3.59	compliment	on	for	4.74
50	detract	from	of	3.43	know	as	in	8.17
51	separate	from	of	3.90	satisfy	with	of	7.76
52	range	from	of	4.07	delight	with	of	5.91
53	vary	from	of	4.23	border	on	of	5.56

	German				French			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
54	deviate	from	of	4.18	regard	as	in	11.88
55	miss	from	in	6.07	prevent	from	of	13.60
56	hang	over	on	10.71	interpret	as	in	7.09
57	interpret	as	in	13.54	endow	with	of	6.26
58	enter	into	in	24.97	enter	into	in	14.40
59	fall	within	in	21.29	benefit	from	of	14.01
60	incorporate	into	in	25.10	treat	as	in	10.83
61	integrate	into	in	25.16	receive	from	of	13.06
62	translate	into	in	24.67	protect	from	of	11.79
63	transform	into	in	22.89	arise	from	of	12.37
64	serve	as	for	28.41	suffer	from	of	13.09
65	force	into	in	22.13	learn	from	of	13.85
66	preside	over	in	20.20	equip	with	of	10.79
67	convert	into	in	24.12	hear	from	of	12.91
68	divide	into	in	25.09	perceive	as	in	9.95
69	transpose	into	in	25.08	exclude	from	of	14.23
70	throw	into	in	24.73	remove	from	of	14.01
71	classify	as	in	25.70	charge	with	of	10.32
72	plunge	into	in	25.11	emerge	from	of	13.82
73	breathe	into	in	24.12	derive	from	of	13.70
74	channel	into	in	25.24	transform	into	in	12.84
75	perceive	as	in	31.11	gain	from	of	12.91
76	treat	as	in	44.25	abstain	from	of	14.24
77	regard	as	for	61.73	stem	from	of	12.87
78	know	as	in	51.43	distance	from	of	11.62
79	view	as	in	42.83	range	from	of	12.28
80	describe	as	in	52.92	refrain	from	of	13.57
81					import	from	of	12.57
82					expect	from	of	13.42
83					differ	from	of	13.76
84					divide	into	in	12.68
85					withdraw	from	of	14.30
86					convert	into	in	13.04

	German				French			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
87					detract	from	of	11.97
88					originate	from	of	13.35
89					stop	from	of	13.67
90					worry	about	of	15.34
91					vary	from	of	13.50
92					translate	into	in	16.77
93					separate	from	of	14.12
94					exempt	from	of	14.04
95					emanate	from	of	12.54
96					profit	from	of	13.43
97					depart	from	of	13.44
98					release	from	of	13.34
99					free	from	of	13.80
100					recover	from	of	13.22
101					quote	from	of	13.56
102					escape	from	of	13.29
103					date	from	of	13.06
104					disappear	from	of	14.20
105					talk	about	of	41.46
106					classify	as	in	15.18
107					incorporate	into	in	23.58
108					deviate	from	of	13.91
109					expel	from	of	14.58
110					integrate	into	in	23.64
111					channel	into	in	15.32
112					fall	within	in	23.69
113					force	into	in	18.59
114					throw	into	in	18.19
115					transpose	into	in	21.20
116					serve	as	of	29.08
117					view	as	in	24.84
118					miss	from	in	21.28
119					complain	about	of	24.72

	German				**French**			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
120					describe	as	of	35.08
121					plunge	into	in	23.77
122					breathe	into	in	23.12
123					hang	over	on	36.75
124					preside	over	of	53.81

	Italian				**Polish**			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
1	call	for	of	1.17	talk	about	of	1.40
2	ask	for	of	1.00	vote	in	for	2.16
3	comply	with	of	1.23	ask	for	of	1.61
4	allow	for	of	1.10	allow	for	on	1.17
5	wait	for	of	1.36	look	at	on	2.29
6	receive	from	by	1.47	deprive	of	by	1.00
7	learn	from	by	1.50	concern	with	of	1.37
8	confer	on	in	1.07	wait	for	on	1.47
9	arise	from	by	1.51	hope	for	on	1.24
10	exclude	from	by	1.55	learn	from	with	1.48
11	protect	from	by	1.49	remove	from	with	1.33
12	hear	from	by	1.48	pass	on	in	1.48
13	remove	from	by	1.57	press	for	on	1.14
14	aim	at	in	2.65	schedule	for	on	1.10
15	emerge	from	by	1.52	aim	at	on	2.37
16	inflict	on	in	1.11	confer	on	in	1.13
17	derive	from	by	1.49	fight	for	of	1.62
18	differ	from	by	1.41	regard	as	for	2.02
19	vary	from	by	1.28	compose	of	with	1.10
20	expect	from	by	1.43	discriminate	against	of	1.34
21	abstain	from	by	1.58	decide	on	of	1.93
22	refrain	from	by	1.51	depend	on	from	2.17
23	gain	from	by	1.47	avail	of	with	1.09
24	import	from	by	1.41	fill	with	by	1.16
25	stem	from	by	1.51	benefit	from	with	2.24

	Italian				**Polish**			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
26	blame	for	of	1.49	worry	about	of	1.50
27	withdraw	from	by	1.55	label	of	in	1.22
28	originate	from	by	1.43	escape	from	with	1.20
29	guard	against	by	1.19	congratulate	on	in	3.03
30	separate	from	by	1.53	withdraw	from	with	1.52
31	range	from	by	1.54	burden	with	of	1.16
32	spend	on	for	2.16	dispose	of	in	1.35
33	delight	with	of	1.43	suffer	from	of	2.04
34	distance	from	by	1.50	exclude	from	with	1.98
35	arrange	for	of	1.32	emerge	from	with	2.03
36	exempt	from	by	1.55	derive	from	with	1.98
37	embark	on	in	1.73	originate	from	with	1.64
38	quote	from	by	1.47	gain	from	with	1.83
39	detract	from	by	1.50	arise	from	with	2.26
40	depart	from	by	1.51	exempt	from	with	1.70
41	free	from	by	1.53	report	on	of	2.12
42	escape	from	by	1.48	protect	from	before	2.22
43	resume	at	in	1.46	recover	from	with	1.64
44	release	from	by	1.53	release	from	with	1.69
45	recover	from	by	1.53	stem	from	with	2.11
46	emanate	from	by	1.51	touch	on	in	2.33
47	disappear	from	by	1.58	import	from	with	2.13
48	expel	from	by	1.55	quote	from	with	1.91
49	profit	from	of	1.65	embark	on	in	2.43
50	congratulate	on	for	3.94	legislate	on	in	1.86
51	deviate	from	by	1.51	emanate	from	with	1.92
52	look	at	in	5.26	inflict	on	in	2.01
53	depend	on	by	3.48	date	from	with	1.98
54	concern	with	of	3.13	disappear	from	with	2.14
55	arrive	at	in	3.06	profit	from	with	2.32
56	benefit	from	of	4.01	warn	against	before	2.44
57	miss	from	in	2.20	estimate	at	on	2.33
58	equip	with	of	2.73	expel	from	with	2.28

	Italian				**Polish**			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
59	search	for	of	2.31	last	for	by	2.30
60	direct	at	in	2.87	rid	of	in	2.27
61	suffer	from	of	3.83	cast	on	in	2.50
62	fill	with	of	2.46	consult	on	in	2.93
63	target	at	in	2.80	direct	at	in	3.07
64	care	for	of	2.86	deliver	on	in	3.25
65	satisfy	with	of	3.94	care	for	on	3.28
66	date	from	of	3.42	border	on	of	2.90
67	estimate	at	in	3.63	arrive	at	in	4.64
68	border	on	with	3.41	target	at	for	3.69
69	dispense	with	of	3.61	charge	with	of	3.63
70	compliment	on	for	4.00	guard	against	before	3.30
71	charge	with	of	4.34	level	at	in	3.80
72	endow	with	of	4.08	transpose	into	for	4.43
73	postpone	until	in	4.77	force	into	for	4.62
74	level	at	in	4.60	amend	on	in	5.27
75	prevent	from	of	12.59	expire	on	in	4.91
76	stop	from	of	8.51	resume	at	in	5.76
77	enter	into	in	18.70	preside	over	by	6.05
78	talk	about	of	34.10	hang	over	on	6.12
79	translate	into	in	18.85	incorporate	into	in	11.31
80	transform	into	in	18.16	throw	into	in	10.57
81	incorporate	into	in	23.53	integrate	into	in	15.82
82	integrate	into	in	23.49	channel	into	in	9.94
83	divide	into	in	18.18	compliment	on	for	12.70
84	convert	into	in	18.22	adjourn	on	in	13.50
85	fall	within	in	24.83	equip	with	in	16.28
86	channel	into	in	17.24	postpone	until	for	16.25
87	throw	into	in	20.00	translate	into	on	21.23
88	force	into	in	21.30	enter	into	in	34.39
89	complain	about	of	21.52	complain	about	on	19.82
90	transpose	into	in	23.73	divide	into	on	22.38
91	plunge	into	in	21.51	miss	from	in	24.72

	Italian				Polish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
92	worry	about	for	30.02	transform	into	in	38.20
93	breathe	into	in	23.00	lag	behind	for	33.71
94	hang	over	on	39.44	hide	behind	for	35.06
95	preside	over	of	54.47	endow	with	in	32.02
96	lag	behind	of	1236.62	fall	within	in	53.05
97					convert	into	in	43.54
98					breathe	into	in	41.85
99					plunge	into	in	44.36

	Spanish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR
1	thank	for	by	1.01
2	deal	with	of	1.36
3	call	for	of	1.16
4	ask	for	of	1.15
5	impose	on	in	1.10
6	consist	of	in	1.02
7	pass	on	of	1.08
8	build	on	in	1.09
9	hope	for	of	1.17
10	allow	for	of	1.31
11	wait	for	of	1.50
12	equip	with	of	1.01
13	apologise	for	by	1.04
14	compensate	for	of	1.19
15	think	of	in	1.85
16	concern	with	of	1.66
17	argue	for	of	1.15
18	aim	at	in	2.45
19	base	on	in	3.66
20	congratulate	on	by	2.53
21	deliver	on	in	1.21
22	qualify	for	of	1.11

	Spanish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR
23	pick	on	of	1.12
24	punish	for	by	1.02
25	touch	on	in	1.62
26	arrange	for	of	1.24
27	elaborate	on	in	1.22
28	acquaint	with	of	1.32
29	destine	for	of	1.48
30	inflict	on	in	1.55
31	focus	on	in	4.01
32	place	on	in	3.05
33	confer	on	in	1.89
34	cater	for	of	1.77
35	impact	on	in	1.94
36	direct	at	in	2.37
37	account	for	of	2.81
38	search	for	of	2.04
39	rest	on	in	2.17
40	arrive	at	in	3.18
41	resume	at	in	2.16
42	look	at	in	6.99
43	dwell	on	in	2.51
44	spend	on	in	3.78
45	concentrate	on	in	4.41
46	insist	on	in	4.31
47	rely	on	in	4.00
48	compliment	on	by	2.70
49	blame	for	of	3.15
50	fill	with	of	2.70
51	found	on	in	3.43
52	test	on	in	2.56
53	level	at	in	2.77
54	embark	on	in	3.61
55	dispense	with	of	3.00

	Spanish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR
56	care	for	of	3.45
57	dream	of	with	3.15
58	adjourn	on	of	3.54
59	target	at	in	3.84
60	charge	with	of	4.13
61	centre	on	in	3.97
62	border	on	with	3.93
63	depend	on	of	8.26
64	endow	with	of	3.99
65	prevent	from	of	10.67
66	talk	about	of	16.73
67	import	from	of	7.91
68	receive	from	of	11.57
69	stop	from	of	8.03
70	hear	from	of	9.86
71	suffer	from	of	11.60
72	benefit	from	of	14.27
73	learn	from	of	13.39
74	count	on	with	11.84
75	vary	from	of	9.17
76	hide	behind	after	9.23
77	gain	from	of	10.61
78	protect	from	of	12.63
79	arise	from	of	13.49
80	exclude	from	of	13.92
81	remove	from	of	13.80
82	estimate	at	in	9.64
83	worry	about	by	12.45
84	emerge	from	of	13.89
85	enter	into	in	20.29
86	date	from	of	10.15
87	channel	into	in	9.86
88	stem	from	of	12.84

	Spanish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR
89	expect	from	of	13.19
90	derive	from	of	14.74
91	emanate	from	of	10.72
92	refrain	from	of	13.94
93	withdraw	from	of	14.50
94	differ	from	of	14.46
95	depart	from	of	12.02
96	incorporate	into	in	19.28
97	force	into	in	13.26
98	exempt	from	of	14.05
99	expire	on	of	11.91
100	free	from	of	13.43
101	distance	from	of	14.43
102	separate	from	of	15.17
103	originate	from	of	15.00
104	integrate	into	in	20.53
105	disappear	from	of	13.79
106	detract	from	of	14.03
107	profit	from	of	14.38
108	release	from	of	13.98
109	expel	from	of	13.80
110	escape	from	of	14.00
111	translate	into	in	19.60
112	quote	from	of	14.62
113	transpose	into	in	16.38
114	regard	as	of	29.27
115	breathe	into	in	14.17
116	recover	from	of	15.10
117	know	as	of	24.22
118	deviate	from	of	15.12
119	transform	into	in	20.38
120	preside	over	of	16.28
121	throw	into	in	17.49

	Spanish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR
122	treat	as	in	25.53
123	miss	from	in	17.67
124	convert	into	in	20.27
125	divide	into	in	20.52
126	abstain	from	in	24.74
127	complain	about	of	20.82
128	plunge	into	in	19.72
129	perceive	as	in	24.19
130	view	as	in	33.18
131	fall	within	in	38.93
132	hang	over	on	32.77
133	interpret	as	in	37.55
134	describe	as	of	71.28
135	serve	as	of	79.84
136	classify	as	of	57.02
137	lag	behind	of	3448.76

C.2.2 Adjective Preposition Combinations

	German				French			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
1	aware	of	on	1.01	responsible	for	of	4.70
2	supportive	of	for	1.22	grateful	for	of	1.01
3	incumbent	on	for	1.10	commensurate	with	of	1.22
4	other	than	on	3.30	eligible	for	of	1.76
5	reminiscent	of	on	1.18	sceptical	about	on	1.81
6	conscious	of	on	1.49	keen	on	of	1.74
7	dependent	on	of	4.47	wrong	with	in	2.38
8	conditional	on	of	1.95	incumbent	on	in	2.20
9	afraid	of	before	2.26	happy	with	of	3.63
10	different	from	of	3.87	conditional	on	of	2.72
11	more	than	on	21.94	dependent	on	of	7.10
12	mindful	of	in	2.54	early	as	in	4.37

	German				French			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
13	sceptical	about	on	2.88	different	from	of	13.64
14	proud	of	on	5.01	serious	about	of	10.72
15	exempt	from	of	3.76	other	than	of	23.40
16	true	of	for	5.09	content	with	of	9.09
17	separate	from	of	3.70	synonymous	with	of	10.33
18	indicative	of	for	4.27	exempt	from	of	13.85
19	ashamed	of	for	5.50	absent	from	of	13.66
20	absent	from	in	6.15	separate	from	of	14.04
21	typical	of	for	7.46	more	than	of	140.38
22	bad	than	behind	9.70	same	as	in	21.43
23	less	than	under	19.14	less	than	of	133.55
24	high	than	on	23.82	low	than	of	140.22
25	low	than	under	20.92	further	than	of	143.77
26	further	than	on	23.93	high	than	by	169.72
27	serious	about	with	28.41	bad	than	of	210.40
28	early	as	in	47.22				
29	same	as	with	139.77				

	Italian				Polish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
1	responsible	for	of	2.43	first	of	in	1.07
2	different	from	by	1.38	critical	of	against	1.03
3	eligible	for	of	1.18	proud	of	with	1.13
4	independent	of	by	1.25	dependent	on	from	2.18
5	conditional	on	in	1.19	third	of	with	1.02
6	true	of	for	1.71	worthy	of	on	1.81
7	same	as	in	1.72	capable	of	in	3.87
8	happy	with	of	1.91	short	of	in	1.55
9	exempt	from	by	1.48	serious	about	of	1.56
10	absent	from	by	1.49	devoid	of	from	1.30
11	dependent	on	by	4.07	valid	for	by	1.33
12	separate	from	by	1.56	exempt	from	with	1.63
13	rich	in	of	2.68	afraid	of	before	1.99

	Italian				Polish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR	λ_v	λ_p	$\lambda_{p''}$	BTR
14	keen	on	in	2.70	free	of	from	2.58
15	more	than	of	41.25	sceptical	about	of	1.94
16	wrong	with	in	5.64	conditional	on	from	2.15
17	content	with	of	5.91	true	of	in	3.47
18	serious	about	on	11.39	commensurate	with	for	3.91
19	sceptical	about	on	11.29	guilty	of	for	2.29
20	other	than	by	30.84	supportive	of	for	3.08
21	further	than	of	17.58	wrong	with	in	3.11
22	early	as	in	17.80	incapable	of	in	4.42
23	synonymous	with	of	17.48	independent	of	from	3.69
24	less	than	of	57.92	absent	from	in	5.18
25	high	than	of	55.43	ashamed	of	for	4.95
26	low	than	of	61.10	typical	of	for	13.11
27	bad	than	of	61.40	early	as	in	64.57

	Spanish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR
1	first	of	in	3.45
2	responsible	for	of	6.26
3	grateful	for	by	1.00
4	eligible	for	of	1.46
5	incumbent	on	in	2.15
6	conditional	on	of	3.39
7	dependent	on	of	8.60
8	wrong	with	in	3.40
9	keen	on	in	3.64
10	sceptical	about	on	5.74
11	different	from	of	14.15
12	more	than	of	101.76
13	serious	about	of	13.79
14	separate	from	of	11.22
15	absent	from	of	11.38
16	same	as	with	19.13

	Spanish			
No.	λ_v	λ_p	$\lambda_{p''}$	BTR
17	synonymous	with	of	12.99
18	exempt	from	of	15.22
19	other	than	of	76.29
20	early	as	in	39.85
21	less	than	of	103.36
22	high	than	of	105.27
23	further	than	of	104.52
24	low	than	of	109.34
25	bad	than	of	107.89

Bibliography

Abdul-Rauf, S., M. Fishel, P. Lambert, S. Noubours and R. Sennrich (2012). "Extrinsic Evaluation of Sentence Alignment Systems". In: *Proceedings of the Workshop on Creating Cross-language Resources for Disconnected Languages and Styles (CREDISLAS)*. (Istanbul), pp. 6–10.

Ahrenberg, L. (2012). "Predicting alignment performance". In: *Proceedings of the 4th Swedish Language Technology Conference (SLTC)*. (Lund), pp. 3–4.

Aijmer, K. (2008). "Parallel and comparable corpora". In: *Corpus Linguistics: An International Handbook*. Ed. by A. Lüdeling and M. Kytö. Vol. 1. Walter de Gruyter, pp. 275–291.

Anderwald, L. and B. Szmrecsanyi (2009). "Corpus linguistics and dialectology". In: *Corpus Linguistics: An International Handbook*. Vol. 2. Walter de Gruyter, pp. 1126–1139.

Antonova, A. and A. Misyurev (2011). "Building a Web-based parallel corpus and filtering out machine-translated text". In: *Proceedings of the 4th Workshop on Building and Using Comparable Corpora (BUCC): Comparable Corpora and the Web*. Association for Computational Linguistics (ACL), pp. 136–144.

Artstein, R. (2017). "Inter-annotator Agreement". In: *Handbook of Linguistic Annotation*. Ed. by N. Ide and J. Pustejovsky. Springer, pp. 297–313.

Ayan, N. F. and B. J. Dorr (2006). "Going Beyond AER: An Extensive Analysis of Word Alignments and Their Impact on MT". In: *Proceedings of the 21st International Conference on Computational Linguistics (COLING) & the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*. (Sydney). Stroudsburg, PA, USA, pp. 9–16.

Baffelli, C. (2016). "An Annotation Pipeline for Italian Based on Dependency Parsing". MA thesis. University of Zurich.

Bahdanau, D., K. Cho and Y. Bengio (2015). "Neural Machine Translation by Jointly Learning to Align and Translate". In: *Proceedings of the International Conference on Learning Representations (ICLR)*.

Baker, M. (1996). "Corpus-based Translation Studies: The Challenges that Lie Ahead". In: *Terminology, LSP and Translation: Studies in Language Engineer-*

ing in Honour of Juan C. Sager. Ed. by H. Somers. Vol. 18. John Benjamins Publishing Company, pp. 175–186.

Ballesteros, L. A. (2002). "Cross-Language Retrieval via Transitive Translation". In: *Advances in Information Retrieval*, pp. 203–234.

Ballesteros, M. and J. Nivre (2012). "MaltOptimizer: An Optimization Tool for MaltParser". In: *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL): Demonstrations*. Association for Computational Linguistics (ACL), pp. 58–62.

Banerjee, S. and A. Lavie (2005). "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments". In: *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Vol. 29. Association for Computational Linguistics (ACL). Ann Arbor, Michigan, pp. 65–72.

Bański, P., E. Frick and A. Witt (2016). "Corpus Query Lingua Franca (CQLF)". In: *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC)*. Ed. by N. Calzolari et al.

Baroni, M., S. Bernardini, A. Ferraresi and E. Zanchetta (2009). "The WaCky wide web: a collection of very large linguistically processed web-crawled corpora". In: *Language Resources and Evaluation* 43.3, pp. 209–226.

Bartsch, S. and S. Evert (2014). "Towards a Firthian Notion of Collocation". In: *Network Strategies, Access Structures and Automatic Extraction of Lexicographical Information. 2nd Work Report of the Academic Network Internet Lexicography*. OPAL – Online publizierte Arbeiten zur Linguistik. Ed. by L. L. Abel A, pp. 48–61.

Bartunov, O. and A. Zakirov (2016). *Better Full Text Search in PostgreSQL*. URL: https://www.slideshare.net/ArthurZakirov1/better-full-text-search-in-postgresql.

Beal, M. J. (2003). "Variational Algorithms for Approximate Bayesian Inference". PhD thesis. The Gatsby Computational Neuroscience Unit, University College London.

Booij, G. (1990). "The boundary between morphology and syntax: separable complex verbs in Dutch". In: *Yearbook of Morphology* 3, pp. 45–63.

Borin, L. (2000a). "Pivot alignment". In: *Proceedings of the 12th Nordic Conference of Computational Linguistics (NODALIDA)*, pp. 41–48.

— (2000b). "You'll Take the High Road and I'll Take the Low Road: Using a Third Language to Improve Bilingual Word Alignment". In: *Proceedings of the 18th International Conference on Computational Linguistics (COLING)*. Saarbrücken, pp. 97–103.

Borin, L. and M. Forsberg (2009). "All in the family: A comparison of SALDO and WordNet". In: *Proceedings of the Workshop on WordNets and other Lexical*

Semantic Resources – between Lexical Semantics, Lexicography, Terminology and Formal Ontologies. Vol. 7.

Bosco, C., F. Dell'Orletta, S. Montemagni, M. Sanguinetti and M. Simi (2014). "The Evalita 2014 Dependency Parsing task". In: *Proceedings of the 1st Italian Conference on Computational Linguistics (CLiC-it) & the 4th International Workshop EVALITA*.

Bott, S. and S. Schulte im Walde (2015). "Exploiting Fine-grained Syntactic Transfer Features to Predict the Compositionality of German Particle Verbs". In: *Proceedings of the 11th International Conference on Computational Semantics (IWCS)*. (Lo), pp. 34–39.

Bouma, G., J. Kuhn, B. Schrader, K. Spreyer, M. Butt and T. H. King (2008). "Parallel LFG Grammars on Parallel Corpora: A base for practical triangulation". In: *Proceedings of the Lexical Functional Grammar (LFG) Conference*. (Sydney). International Lexical Functional Grammar Association (ILFGA), pp. 169–189.

Brants, S., S. Dipper, S. Hansen, W. Lezius and G. Smith (2002). "The TIGER treebank". In: *Proceedings of the 1st International Workshop on Treebanks and Linguistic Theories (TLT)*. Vol. 168.

Brants, T. (2000). "TnT: A Statistical Part-of-Speech Tagger". In: *Proceedings of the 6th Conference on Applied Natural Language Processing (ANLP)*. Association for Computational Linguistics (ACL), pp. 224–231.

Braune, F. and A. Fraser (2010). "Improved Unsupervised Sentence Alignment for Symmetrical and Asymmetrical Parallel Corpora". In: *Proceedings of the 23rd International Conference on Computational Linguistics (COLING): Posters*. Association for Computational Linguistics (ACL), pp. 81–89.

Brown, P. F., S. A. Della Pietra, V. J. Della Pietra, M. J. Goldsmith, J. Hajic, R. L. Mercer and S. Mohanty (1993). "But Dictionaries Are Data Too". In: *Proceedings of the Workshop on Human Language Technology (HLT)*. Association for Computational Linguistics (ACL), pp. 202–205.

Brown, P. F., V. J. Della Pietra, S. A. Della Pietra and R. L. Mercer (1993). "The Mathematics of Statistical Machine Translation: Parameter Estimation". In: *Special issue on using large corpora: II* 19.2, pp. 263–311.

Brown, P. F., P. V. Desouza, R. L. Mercer, V. J. Della Pietra and J. C. Lai (1992). "Class-based n-gram models of natural language". In: *Computational Linguistics* 18.4, pp. 467–479.

Brown, P. F., J. C. Lai and R. L. Mercer (1991). "Aligning sentences in parallel corpora". In: *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 169–176.

Callegaro, E. (2017). "Parallel Corpora for the Investigation of (Variable) Article Use in English: A Construction Grammar Approach". PhD thesis. University of Zurich.

Callegaro, E., S. Clematide, M. Hundt and S. Wick (2018). "Variable article use with acronyms and initialisms – a contrastive analysis of English, German and Italian". In: *Languages in Contrast*. Forthcoming.

Cap, F. (2017). "Show Me Your Variance and I Tell You Who You Are –- Deriving Compound Compositionality from Word Alignments". In: *Proceedings of the Workshop on Multiword Expressions (MWE)*.

Chen, S. F. (1993). "Aligning sentences in bilingual corpora using lexical information". In: *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 9–16.

Chen, Y., A. Eisele and M. Kay (2008). "Improving Statistical Machine Translation Efficiency by Triangulation". In: *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)*. (Marrakech). European Language Resources Association (ELRA), pp. 2875–2880.

Chiarcos, C., S. Dipper, M. Götze, U. Leser, A. Lüdeling, J. Ritz and M. Stede (2008). "A Flexible Framework for Integrating Annotations from Different Tools and Tagsets". In: *Proceedings of the 1st International Conference on Global Interoperability for Language Resources (ICGL)*. Vol. 49. 2, pp. 271–293.

Chiarcos, C., J. Ritz and M. Stede (2012). "By all these lovely tokens…Merging Conflicting Tokenizations". In: *Proceedings of the Linguistic Annotation Workshop (LAW)*. Springer, pp. 53–74.

Christ, O. (1994). "A Modular and Flexible Architecture for an Integrated Corpus Query System". In: *Proceedings of the 3rd Conference on Computational Lexicography and Text Research (COMPLEX)*. (Budapest), pp. 23–32.

Chrupała, G., G. Dinu and J. van Genabith (2008). "Learning Morphology with Morfette". In: *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)*. Vol. 8.

Church, K. W. and P. Hanks (1990). "Word Association Norms, Mutual Information, and Lexicography". In: *Computational Linguistics* 16.1, pp. 22–29.

Clackson, J. (2007). *Indo-European linguistics: an introduction*. Cambridge University Press.

Clematide, S. (2015). "Reflections and a Proposal for a Query and Reporting Language for Richly Annotated Multiparallel Corpora". In: *Proceedings of the Workshop on Innovative Corpus Query and Visualization Tools*. 111. Linköping University Electronic Press. Vilnius, Lithuania, pp. 6–16.

Clematide, S., J. Graën and M. Volk (2016). "Multilingwis – A Multilingual Search Tool for Multi-Word Units in Multiparallel Corpora". In: *Computerised and*

Corpus-based Approaches to Phraseology: Monolingual and Multilingual Perspectives – Fraseologia computacional y basada en corpus: perspectivas monolingües y multilingües. Ed. by G. C. Pastor. Geneva: Tradulex, pp. 447–455.

Cohn, T. and M. Lapata (2007). "Machine Translation by Triangulation: Making Effective Use of Multi-Parallel Corpora". In: *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL)*. (Prague). Vol. 45, pp. 728–735.

Collins, M. (2002). "Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. (Philadelphia). Vol. 10. Association for Computational Linguistics (ACL), pp. 1–8.

Collobert, R. and J. Weston (2008). "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning". In: *Proceedings of the 25th International Conference on Machine Learning (ICML)*. ACM, pp. 160–167.

Costa-jussà, M. R. and R. E. Banchs (2011). *Sentence alignment by means of cross-language information retrieval*. INTECH Open Access Publisher.

Crego, J. M., A. Max and F. Yvon (2010). "Local lexical adaptation in machine translation through triangulation: SMT helping SMT". In: *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*. Association for Computational Linguistics (ACL), pp. 232–240.

Cruz Díaz, N. P. and M. J. Maña López (2015). "An Analysis of Biomedical Tokenization: Problems and Strategies". In: *Proceedings of the 6th International Workshop on Health Text Mining and Information Analysis*. Lisbon, Portugal: Association for Computational Linguistics (ACL), pp. 40–49.

Curzan, A. (2009). "Historical corpus linguistics and evidence of language change". In: *Corpus Linguistics: An International Handbook*. Ed. by A. Lüdeling and M. Kytö. Vol. 2. Walter de Gruyter, pp. 1091–1108.

Cutts, M. (2013). *Oxford Guide to Plain English*. Oxford University Press.

Cysouw, M. and B. Wälchli (2007). "Parallel Texts: Using Translational Equivalents in Linguistic Typology". In: *Sprachtypologie & Universalienforschung (STUF)* 60 (2).

Das, D. and S. Petrov (2011). "Unsupervised Part-of-Speech Tagging with Bilingual Graph-Based Projections". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*. Vol. 1. Portland, Oregon, USA, pp. 600–609.

Davies, M. (2005). "The advantage of using relational databases for large corpora: speed, advanced queries, and unlimited annotation". In: *International Journal of Corpus Linguistics* 10.3, pp. 307–334.

Déjean, H. (2000). "How To Evaluate and Compare Tagsets? A Proposal". In: *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC)*.

Dempster, A. P., N. M. Laird and D. B. Rubin (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm". In: *Journal of the Royal Statistical Society*. B (methodological), pp. 1–38.

DeNero, J., A. Bouchard-Côté and D. Klein (2008). "Sampling Alignment Structure under a Bayesian Translation Model". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Honolulu, Hawaii: Association for Computational Linguistics (ACL), pp. 314–323.

DeNero, J., D. Gillick, J. Zhang and D. Klein (2006). "Why Generative Phrase Models Underperform Surface Heuristics". In: *Proceedings of the Workshop on Statistical Machine Translation*. Association for Computational Linguistics (ACL), pp. 31–38.

DeNero, J. and D. Klein (2008). "The Complexity of Phrase Alignment Problems". In: *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*. (Columbus, Ohio). Stroudsburg, PA, USA, pp. 25–28.

Dewell, R. B. (2015). *The Semantics of German Verb Prefixes*. Vol. 49. John Benjamins Publishing.

Diab, M. and P. Resnik (2002). "An Unsupervised Method for Word Sense Tagging using Parallel Corpora". In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 255–262.

Dyer, C., V. Chahuneau and N. A. Smith (2013). "A Simple, Fast, and Effective Reparameterization of IBM Model 2". In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Association for Computational Linguistics (ACL), pp. 644–649.

Džeroski, S., T. Erjavec and J. Zavrel (2000). "Morphosyntactic Tagging of Slovene: Evaluating Taggers and Tagsets". In: *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC)*.

Einarsson, J. (1976). *Talbankens skriftspråkskonkordans*.

Eisele, A. and Y. Chen (2010). "MultiUN: A Multilingual Corpus from United Nation Documents." In: *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*. (Valletta). European Language Resources Association (ELRA), pp. 2868–2872.

Ejerhed, E. and G. Källgren (1997). *Stockholm Umeå Corpus. Version 1.0*. Tech. rep. Department of Linguistics, Umeå University and Department of Linguistics, Stockholm University.

Erjavec, T., A.-M. Barbu et al. (2010). *MULTEXT-East "1984" annotated corpus 4.0*. URL: https://www.clarin.si/repository/xmlui/handle/11356/1043.

Erjavec, T., D. Fišer, S. Krek and N. Ledinek (2010). "The JOS Linguistically Tagged Corpus of Slovene". In: *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*.

Evert, S. (2004). "The Statistics of Word Cooccurrences: Word Pairs and Collocations". PhD thesis. University of Stuttgart.

— (2008). "Corpora and collocations". In: *Corpus Linguistics. An International Handbook*. Ed. by A. Lüdeling and M. Kytö. Vol. 2. Berlin: Walter de Gruyter, pp. 1212–1248.

Evert, S. and A. Hardie (2011). "Twenty-first century Corpus Workbench: Updating a query architecture for the new millennium". In: *Proceedings of the 5th International Conference on Corpus Linguistics (CL)*. University of Birmingham.

— (2015). "Ziggurat: A new data model and indexing format for large annotated text corpora". In: *Proceedings of the 3rd Workshop on Challenges in the Management of Large Corpora (CMLC)*. (Lancaster). Ed. by P. Bański, H. Biber, E. Breiteneder et al., pp. 21–27.

Firth, J. R. (1957a). "Modes of Meaning". In: *Papers in Linguistics 1934–1951*. London: Oxford University Press.

Firth, J. R. (1957b). "A synopsis of linguistic theory, 1930-1955". In: *Studies in Linguistic Analysis* Special Volume of the Philological Society, pp. 1–31.

Foth, K. (2006). *Eine umfassende Constraint-Dependenz-Grammatik des Deutschen*. Fachbereich Informatik.

Foth, K., A. Köhn, N. Beuck and W. Menzel (2014). "Because Size Does Matter: The Hamburg Dependency Treebank". In: *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*. Fachbereich Informatik.

Fraser, A. and D. Marcu (2007). "Measuring Word Alignment Quality for Statistical Machine Translation". In: *Computational Linguistics* 33.3, pp. 293–303.

Gal, Y. and P. Blunsom (2013). "A Systematic Bayesian Treatment of the IBM Alignment Models". In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Association for Computational Linguistics (ACL), pp. 969–977.

Gale, W. A. and K. W. Church (1991). "A Program for Aligning Sentences in Bilingual Corpora". In: *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*. (Berkeley, California). Stroudsburg, PA, USA, pp. 177–184.

Gale, W. A. and K. W. Church (1993). "A Program for Aligning Sentences in Bilingual Corpora". In: *Computational Linguistics* 19.1, pp. 75–102.

Gamallo, P. and M. Garcia (2013). *FreeLing e TreeTagger: um estudo comparativo no âmbito do Português*. Tech. rep. University of Santiago de Compostela.

Gao, Q. and S. Vogel (2008). "Parallel Implementations of Word Alignment Tool". In: *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*. Association for Computational Linguistics (ACL), pp. 49–57.

Garcia, M. and P. Gamallo (2010). "Análise morfossintáctica para português europeu e galego: Problemas, soluçoes e avaliaçao". In: *Linguamática* 2.2, pp. 59–67.

Gardner, D. and M. Davies (2007). "Pointing Out Frequent Phrasal Verbs: A Corpus-Based Analysis". In: *TESOL quarterly* 41.2, pp. 339–359.

Gilquin, G. and S. Granger (2011). "From EFL to ESL: Evidence from the International Corpus of Learner English". In: *Exploring Second-Language Varieties of English and Learner Englishes: Bridging a Paradigm Gap*, pp. 57–80.

Göhring, A. and M. Volk (2011). "The Text+Berg Corpus An Alpine French-German Parallel Resource". In: *Traitement Automatique des Langues Naturelles*, pp. 63–68.

Goldwater, S. J. (2007). "Nonparametric Bayesian Models of Lexical Acquisition". PhD thesis. Brown University.

Goldwater, S. J., M. Johnson and T. L. Griffiths (2006). "Interpolating Between Types and Tokens by Estimating Power-Law Generators". In: *Advances in Neural Information Processing Systems*, pp. 459–466.

Graën, J. (2017). "Identifying Phrasemes via Interlingual Association Measures". In: *Lexemkombinationen und typisierte Rede im mehrsprachigen Kontext*. Ed. by C. Konecny et al. Tübingen: Stauffenburg Linguistik.

Graën, J., D. Batinic and M. Volk (Oct. 2014). "Cleaning the Europarl Corpus for Linguistic Applications". In: *Proceedings of the Conference on Natural Language Processing (KONVENS)*. (Hildesheim). Stiftung Universität Hildesheim, pp. 222–227.

Graën, J. and C. Bless (2017). "Exploring Properties of Intralingual and Interlingual Association Measures Visually". In: *Proceedings of the 21st Nordic Conference of Computational Linguistics (NODALIDA)*. Linköping Electronic Conference Proceedings 131. Linköping University Electronic Press, Linköpings universitet, pp. 314–317.

Graën, J. and S. Clematide (2015). "Challenges in the Alignment, Management and Exploitation of Large and Richly Annotated Multi-Parallel Corpora". In: *Proceedings of the 3rd Workshop on Challenges in the Management of Large Corpora (CMLC)*. (Lancaster). Ed. by P. Bański, H. Biber et al., pp. 15–20.

Graën, J., S. Clematide and M. Volk (2016). "Efficient Exploration of Translation Variants in Large Multiparallel Corpora Using a Relational Database". In: *Proceedings of the 4th Workshop on Challenges in the Management of Large Corpora (CMLC)*. (Portorož). Ed. by P. Bański, M. Kupietz et al., pp. 20–23.

Graën, J., D. Sandoz and M. Volk (2017). "Multilingwis2 – Explore Your Parallel Corpus". In: *Proceedings of the 21st Nordic Conference of Computational Linguistics (NODALIDA)*. Linköping Electronic Conference Proceedings 131. Linköping University Electronic Press, Linköpings universitet, pp. 247–250.

Graën, J. and G. Schneider (2017). "Crossing the Border Twice: Reimporting Prepositions to Alleviate L1-Specific Transfer Errors". In: *Proceedings of the Joint 6th Workshop on NLP for Computer Assisted Language Learning & 2nd Workshop on NLP for Research on Language Acquisition*. Linköping Electronic Conference Proceedings 134. Linköpings universitet Electronic Press, pp. 18–26.

Granger, S., E. Dagneaux, F. Meunier and M. Paquot (2002). *International Corpus of Learner English*. Presses universitaires de Louvain.

Gries, S. T. (2013). "50-something years of work on collocations: what is or should be next..." In: *International Journal of Corpus Linguistics* 18.1, pp. 137–166.

Grimes, S., K. Peterson and X. Li (2012). "Automatic word alignment tools to scale production of manually aligned parallel texts". English. In: *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*. Ed. by N. Calzolari et al. ACL Anthology Identifier: L12-1265. Istanbul, Turkey: European Language Resources Association (ELRA), pp. 2194–2198.

Gustafson-Capková, S. and B. Hartmann (2006). *Manual of the Stockholm Umeå Corpus version 2.0*. Tech. rep. Department of Linguistics, Stockholm University.

Habert, B., G. Adda, M. Adda-Decker, P. B. de Marëuil, S. Ferrari, O. Ferret, G. Illouz and P. Paroubek (1998). "Towards Tokenization Evaluation". In: *Proceedings of the 1st International Conference on Language Resources and Evaluation (LREC)*. Vol. 98, pp. 427–431.

Habicht, K., H.-J. Kaalep, K. Muischnek, K. Müürisep and A. Rääbis (2000). "Kas tegelik tekst allub eesti keele morfoloogilistele kirjeldustele? Eesti kirjakeele testkorpuse morfosüntaktilise märgendamise kogemusest". In: *Keel ja Kirjandus* 9, pp. 623–633.

Hajlaoui, N., D. Kolovratnik, J. Väyrynen, R. Steinberger and D. Varga (2014). "DCEP-Digital Corpus of the European Parliament". In: *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*, pp. 3164–3171.

Hansen-Schirra, S. (2003). *The Nature of Translated Text - An Interdisciplinary Methodology for the Investigation of the Specific Properties of Translations*. German Research Center for Artificial Intelligence, Saarland University.

Hardie, A. (2012). "CQPweb – combining power, flexibility and usability in a corpus analysis tool". In: *International Journal of Corpus Linguistics* 17.3, pp. 380–409.

Hastings, W. K. (1970). "Monte Carlo Sampling Methods Using Markov Chains and Their Applications". In: *Biometrika* 57.1, pp. 97–109.

Hazel, P. (1997). *PCRE (Perl-compatible regular expressions)*. URL: https://www.pcre.org/.

He, Y. and M. Kayaalp (2006). *A Comparison of 13 Tokenizers on MEDLINE*. Tech. rep. U.S. National Library of Medicine.

Hein, A. S. (2002). "The PLUG project: parallel corpora in Linköping, Uppsala, Göteborg: aims and achievements". In: *Language and Computers* 43.1, pp. 61–78.

Herling, S. H. A. (1821). "Über die Topik der deutschen Sprache". In: *Abhandlungen des frankfurtischen Gelehrtenvereins für deutsche Sprache*, pp. 296–362.

Hermann, K. M. and P. Blunsom (2014). "Multilingual Models for Compositional Distributed Semantics". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 58–68.

Horák, A., L. Gianitsová, M. Šimková, M. Šmotlák and R. Garabík (2004). "Slovak National Corpus". In: *Proceedings of the 7th International Conference on Text, Speech and Dialogue*. (Brno). Springer, pp. 89–93.

ISO/IEC JTC 1 (2016). *ISO/IEC 9075: Information technology – Database languages – SQL*.

Izquierdo, M., K. Hofland and Ø. Reigem (2008). "The ACTRES parallel corpus: an English–Spanish translation corpus". In: *Corpora* 3.1, pp. 31–41.

Jakubíček, M., A. Kilgarriff, V. Kovář, P. Rychlỳ and V. Suchomel (2013). "The TenTen Corpus Family". In: *Proceedings of the 7th International Conference on Corpus Linguistics (CL)*. (Lancaster), pp. 125–127.

Japanese National Institute of Information and Communications Technology (Oct. 17, 2012). *The NICT Japanese Learner English (JLE) Corpus*. URL: https://alaginrc.nict.go.jp/nict_jle/index_E.html.

Junczys-Dowmunt, M., B. Pouliquen and C. Mazenc (2016). "COPPA V2.0: Corpus Of Parallel Patent Applications. Building Large Parallel Corpora with GNU Make". In: *Proceedings of the 4th Workshop on Challenges in the Management of Large Corpora (CMLC)*. (Portorož). Ed. by P. Bański, M. Kupietz et al. Portorož, Slovenia.

Jurish, B. and K.-M. Würzner (2013). "Word and Sentence Tokenization with Hidden Markov Models". In: *Journal for Language Technology and Computational Linguistics (JLCL)* 28.2, pp. 61–83.

Kay, M. (2004). "Substring Alignment using Suffix Trees". In: *Computational Linguistics and Intelligent Text Processing*, pp. 275–282.

Kay, M. and M. Röscheisen (1993). "Text-Translation Alignment". In: *Computational Linguistics* 19.1, pp. 121–142.

Kazakov, D. and A. R. Shahid (2013). "Using Parallel Corpora for Word Sense Disambiguation". In: *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, pp. 336–341.

Kilgarriff, A. (2001). "Comparing Corpora". In: *International Journal of Corpus Linguistics* 6.1, pp. 97–133.

Kilgarriff, A., V. Baisa, J. Bušta, M. Jakubíček, V. Kovář, J. Michelfeit, P. Rychlỳ and V. Suchomel (2014). "The Sketch Engine: Ten Years On". In: *Lexicography* 1.1, pp. 7–36.

Kilgarriff, A. and G. Grefenstette (2003). "Introduction to the Special Issue on the Web as Corpus". In: *Computational Linguistics* 29.3, pp. 333–347.

Kilgarriff, A., M. Husák, K. McAdam, M. Rundell and P. Rychlỳ (2008). "GDEX: Automatically Finding Good Dictionary Examples in a Corpus". In: *Proceedings of the 13th EURALEX International Congress*. Ed. by J. D. Elisenda Bernal. Institut Universitari de Linguistica Aplicada, Universitat Pompeu Fabra.

Kilgarriff, A., P. Rychlỳ, P. Smrž and D. Tugwell (2004). "The Sketch Engine". In: *Information Technology* 105, pp. 116–127.

Klein, W. and A. Geyken (2010). "Das digitale Wörterbuch der deutschen Sprache (DWDS)". In: *Lexicographica: International annual for lexicography*. De Gruyter, pp. 79–96.

Koehn, P. (2005). "Europarl: A parallel corpus for statistical machine translation". In: *Machine Translation Summit*. (Phuket). Vol. 5. Asia-Pacific Association for Machine Translation, pp. 79–86.

— (2010). *Statistical Machine Translation*. Cambridge University Press.

— (May 15, 2012). *European Parliament Proceedings Parallel Corpus 1996-2011*. URL: http://www.statmt.org/europarl/ (visited on Sept. 25, 2017).

Koehn, P., A. Axelrod, A. B. Mayne, C. Callison-Burch, M. Osborne and D. Talbot (2005). "Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation". In: *International Workshop on Spoken Language Translation (IWSLT)*.

Koehn, P., F. J. Och and D. Marcu (2003). "Statistical Phrase-based Translation". In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technolo-*

gies (NAACL-HLT). Vol. 1. Association for Computational Linguistics (ACL), pp. 48–54.

König, E. and W. Lezius (2000). "A Description Language for Syntactically Annotated Corpora". In: *Proceedings of the 18th Conference on Computational Linguistics*. Vol. 2. Association for Computational Linguistics (ACL), pp. 1056–1060.

Krause, T. and A. Zeldes (2014). "ANNIS3: A new architecture for generic corpus query and visualization". In: *Literary and Linguistic Computing* 31.1, pp. 118–139.

Krek, S., K. Dobrovoljc, T. Erjavec, S. Može, N. Ledinek and N. Holz (2015). *Training corpus ssj500k 1.4, Slovenian language resource repository CLARIN.SI*. URL: https://www.clarin.si/repository/xmlui/handle/11356/1052.

Krynicki, G. (2012). "Performance of four sentence aligners on English Polish bitexts". In: *Speech and Language Technology* 14/15.

Kučera, H. and W. N. Francis (1967). *Computational Analysis of Present-Day American English*. Dartmouth Publishing Group.

Kupietz, M., C. Belica, H. Keibel and A. Witt (2010). "The German Reference Corpus DeReKo: A Primordial Sample for Linguistic Research." In: *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*.

Kupietz, M. and H. Lüngen (2014). "Recent Developments in DeReKo". In: *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*. Ed. by N. Calzolari et al. Reykjavik, Iceland: European Language Resources Association (ELRA).

Lafferty, J., A. McCallum and F. C. N. Pereira (2001). "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data". In: *Proceedings of the 18th International Conference on Machine Learning (ICML)*, pp. 282–289.

Langer, S. (2004). "A linguistic test battery for support verb constructions". In: *Lingvisticæ Investigationes* 27.2, pp. 171–184.

Lardilleux, A., J. Gosme and Y. Lepage (2010). "Bilingual lexicon induction: Effortless evaluation of word alignment tools and production of resources for improbable language pairs". In: *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*. European Language Resources Association (ELRA), pp. 252–256.

Lardilleux, A. and Y. Lepage (2007). "The contribution of the notion of hapax legomena to word alignment". In: *Proceedings of the 4th Language and Technology Conference (LTC)*, pp. 458–462.

— (2009). "Sampling-based multilingual alignment". In: *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, pp. 214–218.

Lardilleux, A., Y. Lepage and F. Yvon (2011). "The Contribution of Low Frequencies to Multilingual Sub-sentential Alignment: a Differential Associative Approach". In: *International Journal of Advanced Intelligence* 3.2, pp. 189–217.

Lardilleux, A., F. Yvon and Y. Lepage (2012). "Hierarchical sub-sentential alignment with Anymalign". In: *Proceedings of the 16th Annual Conference of the European Association for Machine Translation (EAMT)*, pp. 279–286.

Lavie, A., A. Parlikar and V. Ambati (2008). "Syntax-driven learning of sub-sentential translation equivalents and translation rules from parsed parallel corpora". In: *Proceedings of the 2nd Workshop on Syntax and Structure in Statistical Translation*. (Ohio). Association for Computational Linguistics (ACL), pp. 87–95.

Lebret, R. and R. Collobert (2014). "Word Embeddings through Hellinger PCA". In: *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. Association for Computational Linguistics (ACL), pp. 482–490.

Leech, G. and A. Wilson (1994). *EAGLES Morphosyntactic Annotation*. Tech. rep. EAG-CSG/IR-T3.

Leech, G. (1992). "100 million words of English: the British National Corpus (BNC)". In: *Language Research* 28.1, pp. 1–13.

Li, Y., L. Xu, F. Tian, L. Jiang, X. Zhong and E. Chen (2015). "Word Embedding Revisited: A New Representation Learning and Explicit Matrix Factorization Perspective." In: *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3650–3656.

Liang, P., B. Taskar and D. Klein (2006). "Alignment by Agreement". In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Association for Computational Linguistics (ACL), pp. 104–111.

Lison, P. and J. Tiedemann (2016). "OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles". In: *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC)*.

Lüdeling, A. (2001). *On Particle Verbs and Similar Constructions in German*. Center for the Study of Language and Information.

Lundborg, J., T. Marek, M. Mettler and M. Volk (2007). "Using the Stockholm TreeAligner". In: *Proceedings of the 6th International Workshop on Treebanks and Linguistic Theorie*.

Ma, X. (2006). "Champollion: A robust parallel text sentence aligner". In: *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, pp. 489–492.

Mann, W. C. and S. A. Thompson (1987). *Rhetorical Structure Theory: A Theory of Text Organization*. University of Southern California, Information Sciences Institute.

— (1988). "Rhetorical structure theory: Toward a functional theory of text organization". In: *Text-Interdisciplinary Journal for the Study of Discourse* 8.3, pp. 243–281.

Manning, C. D., H. Schütze et al. (1999). *Foundations of Statistical Natural Language Processing*. Vol. 999. MIT Press.

Marcu, D. and W. Wong (2002). "A phrase-based, joint probability model for statistical machine translation". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Vol. 10. Association for Computational Linguistics (ACL), pp. 133–139.

Marcus, M. P., M. A. Marcinkiewicz and B. Santorini (1993). "Building a Large Annotated Corpus of English: The Penn Treebank". In: *Computational Linguistics* 19.2, pp. 313–330.

Marneffe, M.-C. de, T. Dozat, N. Silveira, K. Haverinen, F. Ginter, J. Nivre and C. D. Manning (2014). "Universal Stanford Dependencies: A cross-linguistic typology". In: *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*. Ed. by N. Calzolari et al. Vol. 14. European Language Resources Association (ELRA), pp. 4585–4592.

Marneffe, M.-C. de and C. D. Manning (2008). *Stanford typed dependencies manual*. Tech. rep. Stanford University.

Matusov, E., R. Zens and H. Ney (2004). "Symmetric Word Alignments for Statistical Machine Translation". In: *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*. Association for Computational Linguistics (ACL), pp. 219–225.

Mayer, T. and M. Cysouw (2012). "Language comparison through sparse multilingual word alignment". In: *Proceedings of the Joint Workshop of Visualization of Linguistic Patterns (LINGVIS) & Uncovering Language History from Multilingual Resources (UNCLH)*. Association for Computational Linguistics (ACL), pp. 54–62.

McDonald, R. T. et al. (2013). "Universal Dependency Annotation for Multilingual Parsing". In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 92–97.

McEnery, T., A. Wilson, F. Sanchez-Leon and A. Nieto-Serrano (1997). "Multilingual Resources for European Languages: Contributions of the CRATER Project". In: *Literary and Linguistic Computing* 12.4, pp. 219–226.

Medeiros Caseli, H. M. de, C. Ramisch, M. d. G. V. Nunes and A. Villavicencio (2010). "Alignment-based extraction of multiword expressions". In: *Language resources and evaluation* 44.1-2, pp. 59–77.

Mel'čuk, I. (1995). "Phrasemes in language and phraseology in linguistics". In: *Idioms: Structural and psychological perspectives*, pp. 167–232.

— (1998). "Collocations and Lexical Functions". In: *Phraseology. Theory, Analysis, and Applications*. Ed. by A. P. Cowie, pp. 23–53.

Melamed, I. D. (1998a). *Manual Annotation of Translational Equivalence: The Blinker Project*. Tech. rep. University of Pennsylvania.

Melamed, I. D. (1997a). "Automatic Discovery of Non-Compositional Compounds in Parallel Data". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

— (1997b). "Measuring Semantic Entropy". In: *Proceedings of the Special Interest Group on the Lexicon (SIGLEX) Workshop On Tagging Text With Lexical Semantics: Why What And How?*, pp. 4–5.

— (1998b). *Annotation Style Guide for the Blinker Project*. Tech. rep., pp. 58–68.

— (2001). *Empirical Methods for Exploiting Parallel Texts*. MIT press.

Mermer, C. and M. Saraçlar (2011). "Bayesian Word Alignment for Statistical Machine Translation". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*. Vol. 2, pp. 182–187.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller (1953). "Equation of State Calculations by Fast Computing Machines". In: *The Journal of Chemical Physics* 21.6, pp. 1087–1092.

Meurers, W. D. and S. Müller (2009). "Corpora and syntax". In: *Corpus Linguistics: An International Handbook*. Vol. 2. Walter de Gruyter, pp. 920–933.

Michelbacher, L., S. Evert and H. Schütze (2007). "Asymmetric Association Measures". In: *Proceedings of Recent Advances in Natural Language Processing (RANLP)*.

Mihalcea, R. and T. Pedersen (2003). "An Evaluation Exercise for Word Alignment". In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Workshop on Building and using parallel texts: data driven machine translation and beyond*. Vol. 3. Association for Computational Linguistics (ACL), pp. 1–10.

Minh, D. D. L. and D. L. Minh (2015). "Understanding the Hastings algorithm". In: *Communications in Statistics-Simulation and Computation* 44.2, pp. 332–349.

Moore, R. C. (2002). "Fast and accurate sentence alignment of bilingual corpora". In: *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas (AMTA)*. Springer, pp. 135–144.

— (2004). "Improving IBM Word-Alignment Model 1". In: *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 518–525.

Müller, S. (2003). "Solving the bracketing paradox: an analysis of the morphology of German particle verbs". In: *Journal of Linguistics* 39.2, pp. 275–325.

Napoles, C., M. Gormley and B. Van Durme (2012). "Annotated Gigaword". In: *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*. Association for Computational Linguistics (ACL). Association for Computational Linguistics (ACL), pp. 95–100.

Neal, R. M. (1993). *Probabilistic Inference Using Markov Chain Monte Carlo Methods*. Tech. rep. Department of Computer Science, University of Toronto.

Nilsson, J. and J. Hall (2005). *Reconstruction of the Swedish Treebank Talbanken. MSI report 05067*. Tech. rep. Växjö University: School of Mathematics and Systems Engineering.

Nivre, J., J. Hall and J. Nilsson (2006). "MaltParser: A Data-Driven Parser-Generator for Dependency Parsing". In: *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*. Vol. 6, pp. 2216–2219.

Nivre, J. and B. Megyesi (2007). "Bootstrapping a Swedish Treebank Using Cross-Corpus Harmonization and Annotation Projection". In: *Proceedings of the 5th International Workshop on Treebanks and Linguistic Theorie*.

Och, F. J. (1999). "An Efficient Method for Determining Bilingual Word Classes". In: *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. Association for Computational Linguistics, pp. 71–76.

Och, F. J. and H. Ney (2000). "A Comparison of Alignment Models for Statistical Machine Translation". In: *Proceedings of the 18th Conference on Computational Linguistics*. Vol. 2. Association for Computational Linguistics (ACL), pp. 1086–1090.

— (2003). "A Systematic Comparison of Various Statistical Alignment Models". In: *Computational Linguistics* 29.1, pp. 19–51.

Al-Onaizan, Y. et al. (1999). *Statistical Machine Translation*. Tech. rep. Johns Hopkins University Summer Workshop.

Östling, R. (2012). "Stagger: A modern POS tagger for Swedish". In: *Proceedings of the 4th Swedish Language Technology Conference (SLTC)*.

— (2013). "Stagger: An Open-Source Part of Speech Tagger for Swedish". In: *Northern European Journal of Language Technology (NEJLT)* 3, pp. 1–18.

— (2014). "Bayesian Word Alignment for Massively Parallel Texts". In: *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. Association for Computational Linguistics (ACL), pp. 123–127.

— (2015). "Bayesian Models for Multilingual Word Alignment". PhD thesis. Stockholm University.

Östling, R. and J. Tiedemann (2016). "Efficient word alignment with Markov Chain Monte Carlo". In: *Prague Bulletin of Mathematical Linguistics* 106, pp. 125–146.

Papineni, K., S. Roukos, T. Ward and W.-J. Zhu (2002). "BLEU: a Method for Automatic Evaluation of Machine Translation". In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 311–318.

Parker, R., D. Graff, J. Kong, K. Chen and K. Maeda (2011). *English Gigaword Fifth Edition LDC2011T07*. Linguistic Data Consortium. URL: https://catalog.ldc.upenn.edu/LDC2011T07.

Patejuk, A. and A. Przepiórkowski (2010). "ISOcat Definition of the National Corpus of Polish Tagset". In: *Proceedings of the Workshop on Language Resource and Language Technology Standards – state of the art, emerging needs, and future developments*. Ed. by G. Budin, L. Romary, T. Declerck and P. Wittenburg, pp. 23–26.

Petrov, S., D. Das and R. McDonald (2012). "A Universal Part-of-Speech Tagset". In: *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*. Ed. by N. Calzolari et al. Istanbul: European Language Resources Association (ELRA).

Pilán, I., E. Volodina and L. Borin (2016). "Candidate sentence selection for language learning exercises: from a comprehensive framework to an empirical evaluation". In: *Traitement Automatique des Langues* 57.3, pp. 67–91.

Pinker, S. (1996). *Language Learnability and Language Development*. Harvard University Press.

Plamada, M. and M. Volk (Aug. 2013). "Mining for Domain-specific Parallel Text from Wikipedia". In: *Proceedings of the 6th Workshop on Building and Using Comparable Corpora (BUCC)*. (Sofia). Association for Computational Linguistics (ACL), pp. 112–120.

Porter, M. F. (2001). *Snowball: A language for stemming algorithms*. URL: http://snowballstem.org/.

Porter, M. F. (1980). "An algorithm for suffix stripping". In: *Readings in information retrieval*. Vol. 14. 3. Morgan Kaufmann Publishers Inc, pp. 130–137.

PostgreSQL Global Development Group (2017). *PostgreSQL 10 Documentation – Chapter 12. Full Text Search*. URL: https://www.postgresql.org/docs/10/static/textsearch.html.

Przepiórkowski, A., R. L. Górski, B. Lewandowska-Tomaszyk and M. Lazinski (2008). "Towards the National Corpus of Polish". In: *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)*.

Rafalovitch, A., R. Dale et al. (2009). "United Nations General Assembly Resolutions: A Six-Language Parallel Corpus". In: *Proceedings of the Machine Translation Summit*. Vol. 12, pp. 292–299.

Ribeiro, A., G. Lopes and J. Mexia (2000). "Using confidence bands for parallel texts alignment". In: *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 432–439.

Riley, D. and D. Gildea (2010). *Improving the Performance of GIZA++ Using Variational Bayes*. Tech. rep. The University of Rochester, Computer Science Department.

— (2012). "Improving the IBM Alignment Models Using Variational Bayes". In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*. (Jeju Island). Vol. 2. Stroudsburg, PA, USA, pp. 306–310.

Rios, A. (2015). "A Basic Language Technology Toolkit for Quechua". PhD thesis. University of Zurich.

Roberts, G. O., A. Gelman, W. R. Gilks et al. (1997). "Weak Convergence and Optimal Scaling of Random Walk Metropolis Algorithms". In: *The Annals of Applied Probability* 7.1, pp. 110–120.

Rosenfeld, V. (2010). "An Implementation Of The Annis 2 Query Language". MA thesis. Humboldt University of Berlin.

Roßdeutscher, A. (2011). "Particle Verbs and Prefix Verbs in German: Linking Theory versus Word-syntax". In: *Leuvense Bijdragen* 97, pp. 1–53.

Samuelsson, Y. and M. Volk (2006). "Phrase Alignment in Parallel Treebanks". In: *Proceedings of the 5th International Workshop on Treebanks and Linguistic Theories (TLT)*, pp. 91–102.

— (2007). "Alignment Tools for Parallel Treebanks". In: *Proceedings of GLDV Frühjahrstagung*.

Samuelsson, Y., M. Volk, S. Gustafson-Capková and E. J. Steiner (2009). *Alignment Guidelines for SMULTRON*. Tech. rep. Stockholm University, Department of Linguistics.

Santorini, B. (1990). *Part-of-speech Tagging Guidelines for the Penn Treebank, 3rd Revision*. Tech. rep. Department of Computer Science, University of Pennsylvania.

Schiller, A., S. Teufel, C. Stöckert and C. Thielen (1995). *Vorläufige Guidelines fur das Tagging deutscher Textcorpora mit STTS*. Tech. rep. Institute for Natural Language Processing, University of Stuttgart & Department of Linguistics, University of Tübingen.

Schmid, H. (1994). "Probabilistic part-of-speech tagging using decision trees". In: *Proceedings of International Conference on New Methods in Natural Language Processing*. (Manchester). Vol. 12, pp. 44–49.

— (1995). “Improvements in Part-of-Speech Tagging with an Application to German”. In: *In Proceedings of the Workshop of the Special Interest Group on Linguistic data and corpus-based approaches to NLP (SIGDAT)*. (Dublin). Association for Computational Linguistics (ACL).

Schmid, H., A. Fitschen and U. Heid (2004). “SMOR: A German Computational Morphology Covering Derivation, Composition and Inflection.” In: *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*.

Schneider, G. and G. Gilquin (2016). “Detecting innovations in a parsed corpus of learner English”. In: *International Journal of Learner Corpus Research* 2.2, pp. 177–204.

Sennrich, R. and M. Volk (2010). “MT-based sentence alignment for OCR-generated parallel texts”. In: *Proceedings of the 9th Conference of the Association for Machine Translation in the Americas (AMTA)*.

Sennrich, R., M. Volk and G. Schneider (2013). “Exploiting Synergies Between Open Resources for German Dependency Parsing, POS-tagging, and Morphological Analysis”. In: *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, pp. 601–609.

Shannon, C. E. (1948). “A Mathematical Theory of Communication”. In: *The Bell System Technical Journal* 27, pp. 379–423.

Sherlock, C., P. Fearnhead and G. O. Roberts (2010). “The Random Walk Metropolis: Linking Theory and Practice Through a Case Study”. In: *Statistical Science*, pp. 172–190.

Simard, M. (1999). “Text-Translation Alignment: Three Languages Are Better Than Two”. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Simard, M., G. F. Foster and P. Isabelle (1993). “Using Cognates to Align Sentences in Bilingual Corpora”. In: *Proceedings of the Conference of the Centre for Advanced Studies on Collaborative Research (CASCON): Distributed Computing*. Vol. 2. Toronto, Ontario, Canada: IBM Press, pp. 1071–1082.

Simard, M. and P. Plamondon (1998). “Bilingual Sentence Alignment: Balancing Robustness and Accuracy”. In: *Machine Translation* 13.1, pp. 59–80.

Simov, K., P. Osenova and M. Slavcheva (2004). *BTB-TR03: BulTreeBank Morphosyntactic Tagset. BTB-TS version 2.0*. Tech. rep. Bulgarian Academy of Sciences, Sofia, Bulgaria.

Singh, A. K. and S. Husain (2005). “Comparison, Selection and Use of Sentence Alignment Algorithms for New Language Pairs”. In: *Proceedings of the Workshop on Building and Using Parallel Texts*. Association for Computational Linguistics (ACL), pp. 99–106.

Singh, S. (1999). *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography*. Fourth Estate.

Smith, G. (2010). *PostgreSQL 9.0: High Performance*. Packt Publishing Ltd.

Snover, M., B. J. Dorr, R. Schwartz, L. Micciulla and J. Makhoul (2006). "A Study of Translation Edit Rate with Targeted Human Annotation". In: *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas (AMTA)*. Vol. 200. 6.

Søgaard, A. and J. Kuhn (2009). "Empirical lower bounds on alignment error rates in syntax-based machine translation". In: *Proceedings of the 3rd Workshop on Syntax and Structure in Statistical Translation*. Association for Computational Linguistics (ACL), pp. 19–27.

Steinberger, J. et al. (2012). "Creating Sentiment Dictionaries via Triangulation". In: *Decision Support Systems* 53.4, pp. 689–694.

Steinberger, R., M. Ebrahim, A. Poulis, M. Carrasco-Benitez, P. Schlüter, M. Przybyszewski and S. Gilbro (2014). "An overview of the European Union's highly multilingual parallel corpora". In: *Language Resources and Evaluation* 48.4, pp. 679–707.

Steinberger, R., B. Pouliquen, A. Widiger, C. Ignat, T. Erjavec, D. Tufiş and D. Varga (2006). "The JRC-Acquis: A Multilingual Aligned Parallel Corpus with 20+ Languages". In: *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*.

Swanepoel, P. (1998). "Back to basics: prepositions, schema theory, and the explanatory function of the dictionary". In: *Proceedings of the 8th EURALEX International Congress*. Ed. by T. Fontenelle, P. Hiligsmann, A. Michiels, A. Moulin and S. Theissen. Liège, Belgium: Euralex, pp. 655–666.

Teh, Y. W. (2006). "A Hierarchical Bayesian Language Model based on Pitman-Yor Processes". In: *Proceedings of the 21st International Conference on Computational Linguistics (COLING) & the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*. (Sydney), pp. 985–992.

Teleman, U. (1974). *Manual för grammatisk beskrivning av talad och skriven svenska*. Studentlitteratur.

Telljohann, H., E. Hinrichs and S. Kübler (2004). "The TüBa-D/Z Treebank: Annotating German with a Context-Free Backbone". In: *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*.

Telljohann, H., E. Hinrichs, S. Kübler, H. Zinsmeister and K. Beck (2003). *Stylebook for the Tübingen treebank of written German (TüBa-D/Z)*. Tech. rep. Department of Linguistics, University of Tübingen.

The Unicode Consortium (2017). *The Unicode Standard, Version 10.0*.

Tiedemann, J. (2000). "Word Alignment Step by Step". In: *Proceedings of the 12th Nordic Conference of Computational Linguistics (NODALIDA)*, pp. 216–227.

— (2002). "Uplug – a modular corpus tool for parallel corpora". In: *Parallel Corpora, Parallel Worlds* 43. Ed. by L. Borin, pp. 181–197.
— (2003a). "Combining Clues for Word Alignment". In: *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. Vol. 1. Association for Computational Linguistics (ACL), pp. 339–346.
— (2003b). "Recycling Translations: Extraction of Lexical Data from Parallel Corpora and their Application in Natural Language Processing". PhD thesis. Uppsala University.
— (2004). "Word to word alignment strategies". In: *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*. Association for Computational Linguistics (ACL), pp. 212–218.
— (2009). "News from OPUS – A collection of multilingual parallel corpora with tools and interfaces". In: *Proceedings of Recent Advances in Natural Language Processing (RANLP)*. Vol. 5, pp. 237–248.
— (2010). "Lingua-Align: An Experimental Toolbox for Automatic Tree-to-Tree Alignment." In: *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*.
— (2011). *Bitext Alignment*. Vol. 4. Synthesis Lectures on Human Language Technologies 2. Morgan & Claypool.
— (2012). "Parallel Data, Tools and Interfaces in OPUS". In: *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*. (Istanbul).
Tiedemann, J. and G. Kotzé (2009a). "A Discriminative Approach to Tree Alignment". In: *Proceedings of the Workshop on Natural Language Processing Methods and Corpora in Translation, Lexicography, and Language Learning*. Association for Computational Linguistics (ACL), pp. 33–39.
— (2009b). "Building a Large Machine-Aligned Parallel Treebank". In: *Proceedings of the 8th International Workshop on Treebanks and Linguistic Theories (TLT)*, pp. 197–208.
Torres-Ramos, S. and R. E. Garay-Quezada (2015). "A Survey on Statistical-based Parallel Corpus Alignment". In: *Research in Computing Science* 90, pp. 57–76.
Vanallemeersch, T. (2010). "Belgisch Staatsblad Corpus: Retrieving French-Dutch Sentences from Official Documents". In: *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*, pp. 3413–3416.
Varga, D., L. Németh, P. Halácsy, A. Kornai, V. Trón and V. Nagy (2005). "Parallel corpora for medium density languages". In: *Proceedings of Recent Advances in Natural Language Processing (RANLP)*. (Borovets), pp. 590–596.
Vargas, N., C. Ramisch and H. Caseli (2017). "Discovering Light Verb Constructions and their Translations from Parallel Corpora without Word Alignment".

In: *Proceedings of the 13th Workshop on Multiword Expressions (MWE)*, pp. 91–96.

Vilar, D., M. Popovic and H. Ney (2006). "AER: do we need to "improve" our alignments?" In: *International Workshop on Spoken Language Translation (IWSLT)*, pp. 205–212.

Villada Moirón, B. and J. Tiedemann (2006). "Identifying idiomatic expressions using automatic word-alignment". In: *Proceedings of the Workshop on Multiword Expressions in a Multilingual Context*.

Vinyals, O., L. Kaiser, T. Koo, S. Petrov, I. Sutskever and G. Hinton (2015). "Grammar As a Foreign Language". In: *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS)*. Cambridge, MA, USA: MIT Press, pp. 2773–2781.

Vogel, S., H. Ney and C. Tillmann (1996). "HMM-based Word Alignment in Statistical Translation". In: *Proceedings of the 16th Conference on Computational Linguistics*. Vol. 2. Association for Computational Linguistics (ACL), pp. 836–841.

Volk, M., C. Amrhein, N. Aepli, M. Müller and P. Ströbel (2016). "Building a Parallel Corpus on the World's Oldest Banking Magazine". In: *Proceedings of the Conference on Natural Language Processing (KONVENS)*. (Bochum).

Volk, M., S. Clematide, J. Graën and P. Ströbel (2016). "Bi-particle Adverbs, PoS-Tagging and the Recognition of German Separable Prefix Verbs". In: *Proceedings of the Conference on Natural Language Processing (KONVENS)*. (Bochum).

Volk, M. and J. Graën (2017). "Multi-word Adverbs – How well are they handled in Parsing and Machine Translation?" In: *The 3rd Workshop on Multi-word Units in Machine Translation and Translation Technology (MUMTTT)*.

Volk, M., J. Graën and E. Callegaro (2014). "Innovations in Parallel Corpus Search Tools". In: *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*. (Reykjavik). Ed. by N. Calzolari et al. European Language Resources Association (ELRA), pp. 3172–3178.

Volk, M., J. Lundborg and M. Mettler (2007). "A Search Tool for Parallel Treebanks". In: *Proceedings of the Linguistic Annotation Workshop (LAW)*. Prague: Association for Computational Linguistics (ACL), pp. 85–92.

Volodina, E., R. Johansson and S. J. Kokkinakis (2012). "Semi-automatic selection of best corpus examples for Swedish: Initial algorithm evaluation". In: *Proceedings of the Workshop on NLP for Computer Assisted Language Learning*. 080. Linköping University Electronic Press, pp. 59–70.

Voutilainen, A., T. Purtonen and K. Muhonen (2012). *FinnTreeBank2 Manual*. Tech. rep. University of Helsinki, Department of Modern Languages.

Wanner, L. (1996). *Lexical Functions in Lexicography and Natural Language Processing*. Vol. 31. John Benjamins Publishing.

Winand, M. (2012). *SQL Performance Explained: Everything Developers Need to Know about SQL Performance*.

Wu, H. and H. Wang (2009). "Revisiting Pivot Language Approach for Machine Translation". In: *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics (ACL) & the 4th International Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (AFNLP)*. Vol. 1, pp. 154–162.

Xiao, R. (2008). "Well-known and influential corpora". In: *Corpus Linguistics: An International Handbook*. Ed. by A. Lüdeling and M. Kytö. Vol. 1. Walter de Gruyter, pp. 383–457.

Yannakoudakis, H., T. Briscoe and B. Medlock (2011). "A New Dataset and Method for Automatically Grading ESOL Texts". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*. Vol. 1, pp. 180–189.

Yu, Q., A. Max and F. Yvon (2012). "Revisiting sentence alignment algorithms for alignment visualization and evaluation". In: *Proceedings of the 5th Workshop on Building and Using Comparable Corpora (BUCC)*. Special Theme: "Language Resources for Machine Translation in Less-Resourced Languages and Domains". (Istanbul), pp. 10–16.

Zarrieß, S. and J. Kuhn (2009). "Exploiting Translational Correspondences for Pattern-Independent MWE Identification". In: *Proceedings of the Workshop on Multiword Expressions (MWE): Identification, Interpretation, Disambiguation and Applications*. Association for Computational Linguistics (ACL), pp. 23–30.

Zeller, J. (2001). "Particle Verbs and Local Domains". In: *Linguistik Aktuell/Linguistics Today* 41.

Zhechev, V. (2009). "Automatic Generation of Parallel Treebanks: An Efficient Unsupervised System". PhD thesis. Dublin City University.

Zhechev, V. and A. Way (2008). "Automatic generation of parallel treebanks". In: *Proceedings of the 22nd International Conference on Computational Linguistics (COLING)*. Vol. 1, pp. 1105–1112.

Zielinski, A., C. Simon and T. Wittl (2009). "Morphisto: Service-Oriented Open Source Morphology for German". In: *State of the Art in Computational Morphology*, pp. 64–75.

Ziemski, M., M. Junczys-Dowmunt and B. Pouliquen (2016). "The United Nations Parallel Corpus v1.0". In: *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC)*. Portorož, Slovenia.

Zipf, G. (1949). "Human Behaviour and the Principle of Least Effort: An Introduction to Human Ecology". In: *Social Forces* 28 (3), pp. 340–341.